

Universidad Carlos III de Madrid

**Ingeniería Industrial: Mecánica de
máquinas y estructuras**



Proyecto fin de carrera

**Diseño y montaje de un prototipo funcional de
un afinador automático de guitarra**

Autor: Javier Sánchez Navasa

Tutor: Jorge Pleite Guerra

Agradecimientos:

A Jose María, que me dio la idea de hacer este proyecto, buenos consejos sobre conseguir condiciones de laboratorio y parte de su laboratorio.

A la empresa EXPACE On Board Systems, por el apoyo humano, (Manuel haciendo las veces de tutor, el departamento de electrónica, sugiriendo componentes y circuitos, y Fran y sus conocimientos de Matlab y Arduino), y material, ya que me han provisto de tantos componentes como he necesitado

A Jose Manuel, que me ha prestado su experiencia sobre fabricación, su tiempo y sus máquinas.

A Juan que ofreció de manera altruista parte de sus vacaciones para ayudarme a programar el microprocesador.

Y a todos aquellos/a que han esperado pacientemente a que terminara y presentara el proyecto.

Índice

1. Introducción:	5
1.1 Estado del arte	5
1.1.1 Transperformance Tuning System	5
1.1.2 Tronical Tune.....	6
1.1.3 Peavey AT 200	7
1.1.4 Line 6 JTV.....	8
1.2 Objetivos del proyecto:	8
2. Solución propuesta.....	10
3. Desarrollo teórico.....	12
3.1 Mecánica	13
3.1.1 Calculo de la Tensión de las cuerdas.....	13
3.1.2 Elección de componentes de la reductora, relación de transmisión, etc... ..	16
3.1.3 Características geométricas de la reductora comercial.	16
3.1.4 Selección del motor.....	18
3.1.5 Calculo de fuerzas en los engranajes.	19
3.1.6 Calculo de la resistencia del engranaje.	21
3.1.7 Calculo de fuerzas en los ejes.....	22
3.1.8 Diagrama de momentos axiles, flectores y torsores.....	26
3.1.9 Calculo de la resistencia a fatiga de los ejes	30
3.1.10 Calculo de la resistencia a fluencia.....	39
3.1.11 Diseño de la unión del motor con el eje del sinfín.....	39
3.2 Electrónica	42
3.2.1 Descripción del sistema y partes.....	42
3.2.2 Alimentación y adquisición y acondicionamiento de la señal:	44
3.2.3 Control del sistema.	46
3.2.4 Interfaz con el usuario.....	47
3.2.5 Drivers y motores.	49
3.3. Software	53
3.3.1. Funcionamiento del programa.....	53
3.3.1.1 Modo Inicio.	53
3.3.1.2 Modo automático:	54
3.3.1.3 Modo manual	55
3.3.2 Modo automático: Base teórica.....	56

4. Diseño en 3D, fabricación, montaje y pruebas	64
4.1 Diseño en 3D.	64
4.2 Fabricación	64
4.2.1 Fabricación de piezas mecánicas	64
4.2.2 Fabricación de PCBs	67
4.2.3 Fabricación de cableado.....	69
4.3 Montaje	69
4.4 Pruebas.....	72
4.4.1 Prueba de LEDs y pulsadores	72
4.4.2 Prueba de los motores	72
4.4.3 Prueba del modo automático	73
4.4.4 Pruebas de control automático de los motores.....	74
5. Recursos.	75
5.1 Coste de material:	75
5.2 Recursos asignados a las tareas.	76
6. Conclusiones:	78
6.1 Evaluación del sistema.	78
6.2 Mejoras al prototipo actual.....	79
6.4 Mejoras diferenciadoras	80
Anexos	82
Anexo A: Propiedades mecánicas de algunos materiales.....	82
Anexo B: Código del afinador en arduino y matlab.	84
B.1 Código del sistema:	84
B.2 Código afinador Matlab	90
Bibliografía	92

1. Introducción:

La afinación de un instrumento es una de las mayores preocupaciones para cualquier músico, especialmente para aquellos que utilicen instrumentos de cuerda como guitarras, bajos, violines, arpas... Incluso la mejor interpretación del mejor músico podría parecer pésima si su instrumento no ha sido afinado correctamente.

Por eso, un guitarrista que toque en directo suele reajustar sus cuerdas después de cada canción. Este proceso de tocar la cuerda, mirar la pantalla del afinador, apretar/aflojar la clavija y volver a comprobar el afinador hasta tener a punto las 6 cuerdas, puede durar bastante tiempo.

Por otro lado, cuando alguien empieza a practicar música no es capaz de darse cuenta de que su instrumento esta desafinado. Lo que toca suena "mal" pero no se está equivocando, es que no ha afinado su instrumento.

Lo que estos músicos necesitan es un aparato que compruebe la afinación de las cuerdas y las afine correctamente en el menor tiempo posible y de manera automática. Y eso, es lo que se ha tratado de conseguir en este proyecto.

1.1 Estado del arte

Algo que la mayoría ignora es que esto no es una idea del todo novedosa. Desde los 90 hasta la actualidad se han ido desarrollando diferentes tipos de soluciones al problema, cada una con sus ventajas y sus inconvenientes, con un precio que oscila entre los 300 y los 3000 euros, y que se basan en incorporar a las guitarras componentes electromecánicos internos y externos, dotados de la tecnología suficiente como para reafinar una guitarra en cuestión de segundos.

1.1.1 Transperformance Tuning System

En 1994 aparece el primer afinador automático de guitarras, llamado Transperformance Tuning System y desde 2008 conocido como Axcent Tuning System.



Figura 1.1 Transperformance Tuning System [1]

Este sistema está compuesto por un ordenador (microprocesador), sensores en las cuerdas, un sistema mecánico con motores paso a paso, 12 botones y una pantalla LCD.

Funciona en lazo abierto y está basado en la precisión de los propios componentes, porque permite tanto afinar la guitarra, como cambiar de tono de las cuerdas sin dejar de tocar

(debido a que tiene almacenados algoritmos que calculan el número de pasos del motor necesarios para llegar a cada tono en función del calibre de la cuerda y el tono de origen).

Es necesario mecanizar la guitarra para la instalación en el instrumento. La propia empresa ofrece un servicio de instalación, mantenimiento y repuesto de piezas para quien quiera disfrutar del sistema.

El precio de una guitarra (comprada directamente a Axent tuning) ronda los 4000\$, ya que únicamente se instala en guitarras de gran calidad, aun así, supone un incremento de casi 3000\$ sobre el valor de la guitarra.

1.1.2 Tronical Tune

En 2005 la compañía Tronical fabrica su propio sistema de auto afinación, llamado 'Power Tune'. Se instala en 2007 en las 'Gibson robot guitar' de edición limitada y de alrededor de 2500\$, por aquel entonces el sistema de auto afinación PowerTune costaba 800 euros. El sistema está compuesto por: servomotores en las clavijas, un puente que analiza el sonido y un potenciómetro selector donde se pueden almacenar varias afinaciones y presets.



Figura 1.2 Gibson Robot Guitar [2]

En 2012, Tronical sacó un nuevo modelo de afinador automático, donde todo el hardware se transportaba a la pala de la guitarra, eliminando el puente con sensores y posibilitando la colocación del autotune a cualquier guitarra, con un coste de aproximadamente 300 euros



Figura 1.3 Tronical Tuning System [3]

Características: Dispositivo acoplado a la pala de la guitarra, es necesario sustituir las clavijas anteriores para colocar las del dispositivo, pero no hay que mecanizar la madera de la guitarra para instalarlo. Valido casi para cualquier modelo de guitarra. Tiene en la memoria 24 diferentes afinaciones (presets), y otras 12 las define el usuario. Afinación rápida y precisa (en algunos segundos).

1.1.3 Peavey AT 200

En 2011 Peavey lanzó al mercado su primera guitarra con un sistema de afinación automática. La principal característica es que no lleva ningún hardware externo, por lo que a primera vista parece una guitarra convencional.

La guitarra cuenta con dos sistemas, el primero, llamado 'Antares autotune' afina la guitarra con apretar un botón, y además, cuenta con un sistema 'Solid Tune Intonation' que comprueba en cada momento la afinación y hace pequeñas modificaciones para que el sonido sea constantemente perfecto.

Si se actualiza el software de Antares, se puede cambiar la afinación de la guitarra al apretar el botón. Puede comprarse por 200 euros.



Figura 1.4 Peavey AT200 [4]

1.1.4 Line 6 JTV

En 2010 Line 6 en colaboración con James Tyler (luthier) crearon una guitarra 'Line 6 JTV' que no solo se afina sola, cambia de afinación, etc. Si no que además, gracias a la tecnología MIDI, ofrece la posibilidad de cambiar del sonido usual de la guitarra eléctrica a cualquier otro tipo de guitarra: acústica, de jazz... Igual que en el caso anterior, el hardware se sitúa en el interior de la guitarra.



Figura 1.5 Line 6 JTV [5]

A modo de resumen:

SISTEMA	VENTAJAS	INCONVENIENTES
Axcent Tuning System	Permite cambiar la afinación sin dejar de tocar.	Precio alrededor de 3000 \$. Necesita mecanizar el instrumento para su instalación
Tronical	Precio alrededor de 300 \$ Válido para todos los modelos usuales de guitarra (LP,ST,SG,TC)	Hardware a la vista.
Peavey AT 200	Mantiene la afinación en todo momento, incluso tocando	No se puede incorporar en otro modelo de guitarra
Line 6 JTV	Alta tecnología, muchas posibilidades de sonido	Pérdida de identidad de lo que debe ser una guitarra eléctrica.

El sistema que se quiere construir va a ser parecido al desarrollado por Tronical, en cuanto a la localización del hardware, y se ha usado como referencia a la hora de limitar los costes de material del proyecto. Sin embargo, el algoritmo de afinación es más parecido al de Axcent Tuning System

1.2 Objetivos del proyecto:

El objetivo de este proyecto adquirir el know-how sobre afinadores automáticos y construir un **prototipo funcional**. Una vez construido se evaluarán los puntos fuertes y debilidades del mismo, para decidir si sería rentable seguir investigando y gastando recursos en el desarrollo de un producto. Para ello se van a definir una serie de objetivos secundarios y requisitos que debe cumplir el afinador:

Objetivos:

- Encontrar una solución de bajo coste que permita acoplar los motores a la reductora de las clavijas de la guitarra.
- Diseñar los elementos mecánicos del sistema lo más sencillos posibles, mientras sean funcionales, para abaratar costes.
- Conocer el funcionamiento de las diferentes soluciones que ofrece el mercado para capturar el sonido con el fin de decidir cuál es la óptima para este sistema.
- Comprender el funcionamiento de la transformada discreta de Fourier y aplicarla para calcular la frecuencia fundamental del sonido.
- Diseñar un algoritmo de control que permita afinar con precisión en el menor tiempo posible, puede ser tanto en bucle abierto como en bucle cerrado.
- Calcular la viabilidad económica de la construcción de un sistema de afinación automática de guitarras.

El prototipo debe cumplir los siguientes requisitos:

- Debe afinar la guitarra de forma automática. El usuario pulsará las cuerdas de una en una y el sistema las irá afinado consecutivamente.
- Debe avisar al usuario cuando las cuerdas estén afinadas.
- La guitarra no debe perder funcionalidad una vez se haya instalado el sistema. Además, en la instalación la madera de la guitarra no debe sufrir modificaciones (mecanizados, taladros etc.)
- El sistema se debe alimentar con baterías o pilas, debido a que la red introduce ruidos en el sistema que pueden alterar la afinación.
- Coste máximo del material a utilizar 300 euros.
- El número de piezas no comerciales incorporadas debe ser el mínimo, ya que encarecen el sistema.

Dado que el prototipo objetivo se aleja de cualquier producto de mercado y no se pretende comercializarlo, no se han considerado restricciones de tamaño y forma porque encarecen la fabricación y dificultan la búsqueda y compra de componentes comerciales. Tampoco se van a tener en cuenta cuestiones estéticas, así que no es necesario introducir la electrónica dentro de carcasas u ocultar cables. En resumen, se prescinde de todo lo que no sea necesario para el funcionamiento del sistema.

2. Solución propuesta.

Se van a plantear los diferentes bloques con los que va a contar el afinador automático. Son los que se creen necesarios para que el sistema funcione correctamente:

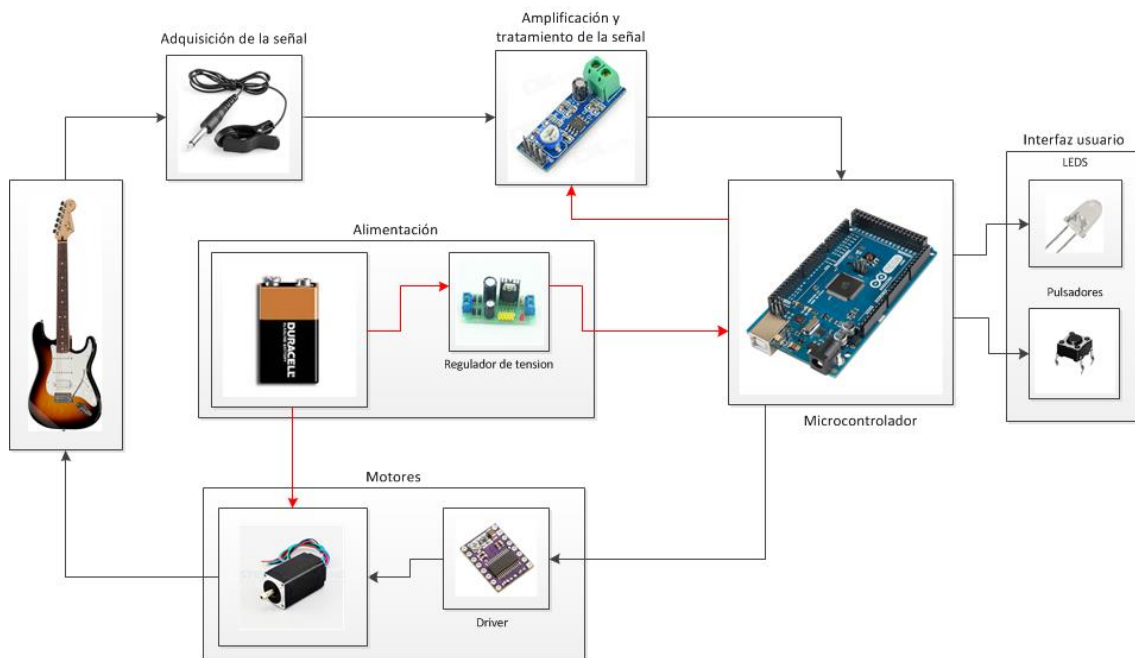


Figura 2.0.1 Esquema básico del sistema

El sistema se compone de los siguientes bloques:

- **Adquisición de la señal:**
Tiene la misión de capturar la vibración producida por las cuerdas de la guitarra. Se debe decidir entre utilizar un sensor piezoeléctrico, que registraría la vibración de la madera; o un micrófono, que recoge la vibración del aire.
- **Amplificación y acondicionamiento de la señal:**
La señal capturada por los micrófonos y piezoeléctricos es realmente débil, por lo que va a ser necesario amplificarla. Además, dependiendo del microcontrolador habrá acondicionarla para que la pueda leer. Hay que elegir un amplificador y tener en cuenta las especificaciones del ADC del microcontrolador al diseñar el circuito.
- **Control del sistema: Microcontrolador.**
Contiene el programa principal que gobierna el sistema, se encarga de la digitalización de la señal, detección de la frecuencia fundamental del sonido (aplicando la transformada rápida de Fourier), controla los motores y actúa sobre los pulsadores y los LEDs. Se puede utilizar para alimentar el amplificador para que trabaje en el rango de tensiones del microcontrolador.
La decisión entre un microcontrolador u otro está condicionada por su capacidad para realizar la FFT, el número de entradas y salidas analógicas y digitales y su facilidad de programación.

- **Alimentación:**

Tiene que ser independiente de la red. Hay que decidir si se ponen pilas o baterías. La alimentación de los motores generalmente se hace a mayor tensión que la del microcontrolador, habrá que utilizar un regulador de tensión.

- **Interfaz con el usuario:**

La interfaz con el usuario la constituyen un grupo de LEDs y pulsadores. Los leds sirven para mostrar al usuario el estado de la afinación y los pulsadores sirven para interactuar con el programa. Hay que diseñar un circuito electrónico que incluya estos LEDs y pulsadores.

- **Motores:**

Para variar la tensión de las cuerdas y por tanto la afinación se necesitan motores, hay que decidir si usar motores de continua o paso a paso. Para poder reducir su tamaño, van a contar con una reductora mecánica acoplada a su eje.

Debido a la tensión que necesitan los motores para funcionar, no se pueden controlar directamente desde el microcontrolador, así que hay que utilizar un driver.

3. Desarrollo teórico.

En vista de la solución propuesta, se definen tres grandes áreas diferenciadas a estudiar: mecánica, electrónica y software.

- **Mecánica:** Este apartado incluye el cálculo de la tensión de las cuerdas, la elección de una reductora apropiada, dimensionamiento y elección de los motores y cálculos teóricos de la resistencia mecánica de la reductora (engranajes y ejes). Además se expone el 3D del sistema con todos los componentes, para estudiar la mejor manera de colocarlos y se incluye el diseño del sistema de transmisión de potencia de los motores al eje de las clavijas (reductora) y de las piezas necesarias para el anclaje.
- **Electrónica:** Incluye toda la circuitería externa al microcontrolador, la elección y/o diseño de los circuitos de acondicionamiento de la señal, de la interfaz con el usuario, de los drivers para los motores y el regulador de tensión. Además, se expone el esquema de cableado y el placement de las PCBs que haya que fabricar.
- **Software:** Incluye la explicación del algoritmo que se encarga del funcionamiento del sistema, y las pruebas de los diferentes componentes, ya sea para aprender su uso o caracterizar su comportamiento,

Estos bloques se van a desarrollar de manera teórica, justificando las decisiones tomadas y los cálculos realizados.

3.1 Mecánica

Para poder calcular la potencia de los motores y la reductora, lo primero que hay que conocer es la tensión de cada cuerda de la guitarra.

3.1.1 Calculo de la Tensión de las cuerdas.

La tensión de una cuerda depende de su afinación, del calibre (diámetro) de cada cuerda y de la longitud del mástil de la guitarra.

Las guitarras tienen 6 cuerdas, cada una de un diámetro diferente, tres lisas y tres “entorchadas”. El calibre de las cuerdas aumenta desde la más aguda a la más grave para que todas las cuerdas tengan aproximadamente la misma tensión.

Dependiendo del estilo del guitarrista, la afinación y el calibre de las cuerdas de la guitarra varía. Aquí tenemos un ejemplo.

Medidas de cuerda estandards

Nombre	1 (E)	2 (B)	3 (G)	4 (D)	5 (A)	6 (E)
Extra super light (8-38)	0.08	0.10	0.15	0.21	0.30	0.38
Super light (9-42)	0.09	0.11	0.16	0.24	0.32	0.42
Custom light (9-46)	0.09	0.11	0.16	0.26	0.36	0.46
Light (10-46)	0.10	0.13	0.17	0.26	0.36	0.46
Light (10-52)	0.10	0.13	0.17	0.32	0.42	0.52
Medium (11-49)	0.11	0.14	0.18	0.28	0.38	0.49
Heavy (12-52)	0.12	0.16	0.24	0.32	0.42	0.52
Baritone (12-68)	0.12	0.16	0.22	0.38	0.52	0.68

Figura 3.1 Calibre estándar de un juego de cuerdas de guitarra [6]

Los cálculos de este proyecto se van hacer para afinación estándar E-B-G-D-A-E (mi – Si – Sol – Re – La – Mi) que es la que se usa para la mayoría de estilos, con cuerdas de calibres de 9 a 46 (Figura 1). Las longitudes del mástil de la guitarra más comunes son 25,5 pulgadas (65cm) de Fender y 23 pulgadas (63 cm), Gibson. Como veremos más adelante, la longitud del mástil es directamente proporcional a la tensión de las cuerdas, por lo que, para los cálculos, se va a usar el más largo, 25,5 pulgadas, que engloba al de 23.

Existen fórmulas empíricas para el cálculo de la tensión de una cuerda,

- $T = 4 \cdot F^2 \cdot L \cdot M / 980621$ [7] donde T es la tensión en kgs, F la frecuencia en Hz, L la longitud de la cuerda que vibra en centímetros y M la masa de esa longitud de

cuerda en gramos. La constante es el producto de la constante gravitatoria por las operaciones de cambio de unidades.

- T (*Tension*) = $(UW \times (2 \times L \times F)^2) \cdot 4.45 / 386.4$ [8], esta ecuación aparece en el "Catalog Supplement/String Tension Specifications" de D'Addario, (Un fabricante de cuerdas de guitarra). T es la tensión en Newtons, UW el peso por unidad de longitud en libras/pulgada, y F la frecuencia de vibración de la cuerda en Hz, 386.4 es una constante y 4.45 para convertir el resultado a Newtons.

En el mismo catálogo de D'Addario hay tablas con las tensiones de las cuerdas de diferente calibre y tamaño de mástil. Todas han sido calculadas con la segunda fórmula. De esta tabla se pueden obtener las tensiones de las seis cuerdas. (Figura 3.2):

- 1ª cuerda, afinada en Mi: $T = 13,1 \cdot 4.45 = 58,3 N$
- 2ª cuerda, afinada en Si: $T = 11 \cdot 4.45 = 49 N$
- 3ª cuerda, afinada en Sol: $T = 14,7 \cdot 4.45 = 65,4 N$
- 4ª cuerda, afinada en Re: $T = 18,4 \cdot 4.45 = 81,9 N$
- 5ª cuerda, afinada en La: $T = 19,5 \cdot 4.45 = 86,8 N$
- 6ª cuerda, afinada en MI: $T = 17,5 \cdot 4.45 = 77,9 N$

La tensión más alta la soporta la 5ª cuerda, 86,8 Newtons.

Con esta tensión se pueden calcular los engranajes que moverán las clavijas de la guitarra, y elegir el motor y la potencia necesarios para que pueda funcionar el mecanismo.

acoustic or electric guitar

Plain Steel - Lock Twist

Item#	Unit Weight	a'	f'	e'	d'	c'	b	a	a
PL007	.0001085	11.2	8.9	7.9	6.3	5.0	4.5	3.5	2.8
PL008	.0001418	14.7	11.6	10.4	8.2	6.5	5.8	4.6	3.7
PL0085	.0001501	16.6	13.1	11.7	9.3	7.4	6.6	5.2	4.1
PL009	.0001794	18.6	14.7	13.1	10.4	8.3	7.4	5.8	4.6
PL0095	.0001999	20.7	16.4	14.6	11.6	9.2	8.2	6.5	5.2
PL010	.0002215	22.9	18.2	16.2	12.9	10.2	9.1	7.2	5.7
PL0105	.0002442	25.3	20.0	17.9	14.2	11.3	10.0	8.0	6.3
PL011	.0002680	27.7	22.0	19.6	15.6	12.3	11.0	8.7	6.9
PL0115	.0002930	30.3	24.0	21.4	17.0	13.5	12.0	9.5	7.6
PL012	.0003190	33.0	26.2	23.3	18.5	14.7	13.1	10.4	8.2
PL013	.0003744	38.7	30.7	27.4	21.7	17.2	15.4	12.2	9.7
Item#	Unit Weight	a'	c'	b	a	f	e	d	a
PL0135	.0004037	23.4	18.6	16.6	13.2	10.4	8.3	7.4	5.9
PL014	.0004342	25.2	20.0	17.8	14.1	11.2	8.9	7.9	6.3
PL015	.0004984	28.9	23.0	20.5	16.2	12.9	10.2	9.1	7.2
PL016	.0005671	32.9	26.1	23.3	18.5	14.7	11.6	10.4	8.2
PL017	.0006402	37.2	29.5	26.3	20.9	16.6	13.1	11.7	9.3
Item#	Unit Weight	b	a	a	f	e	d	e	B
PL018	.0007177	29.5	23.4	18.6	14.7	13.1	10.4	8.3	7.4
PL019	.0007997	32.8	26.1	20.7	16.4	14.6	11.6	9.2	8.2
PL020	.0008861	36.4	28.9	22.9	18.2	16.2	12.9	10.2	9.1
Item#	Unit Weight	a	a	f	e	d	e	B	A
PL022	.0010722	34.9	27.7	22.0	19.6	15.6	12.3	11.0	8.7
PL024	.0012760	41.6	33.0	26.2	23.3	18.5	14.7	13.1	10.4
PL026	.0014975	48.8	38.7	30.7	27.4	21.7	17.2	15.4	12.2

Plain Steel - Soldered Twist

Item#	Unit Weight	a'	f'	e'	d'	c'	b	a	a
KPL009	.0001794	18.6	14.7	13.1	10.4	8.3	7.4	5.8	4.6
KPL010	.0002215	22.9	18.2	16.2	12.9	10.2	9.1	7.2	5.7
KPL011	.0002680	27.7	22.0	19.6	15.6	12.3	11.0	8.7	6.9
KPL012	.0003190	33.0	26.2	23.3	18.5	14.7	13.1	10.4	8.2
KPL013	.0003744	38.7	30.7	27.4	21.7	17.2	15.4	12.2	9.7
Item#	Unit Weight	a'	c'	b	a	f	e	d	a
KPL014	.0004342	25.2	20.0	17.8	14.1	11.2	8.9	7.9	6.3
KPL016	.0005671	32.9	26.1	23.3	18.5	14.7	11.6	10.4	8.2
KPL017	.0006402	37.2	29.5	26.3	20.9	16.6	13.1	11.7	9.3
KPL020	.0008861	51.5	40.8	36.4	28.9	22.9	18.2	16.2	12.9

Plain Steel - Double Ball End

Item#	Unit Weight	a'	f'	e'	d'	c'	b	a	a
SPL009	.0001794	18.6	14.7	13.1	10.4	8.3	7.4	5.8	4.6
SPL010	.0002215	22.9	18.2	16.2	12.9	10.2	9.1	7.2	5.7
SPL011	.0002680	27.7	22.0	19.6	15.6	12.3	11.0	8.7	6.9
SPL013	.0003744	38.7	30.7	27.4	21.7	17.2	15.4	12.2	9.7
Item#	Unit Weight	a'	c'	b	a	f	e	d	a
SPL016	.0005671	32.9	26.1	23.3	18.5	14.7	11.6	10.4	8.2
SPL017	.0006402	37.2	29.5	26.3	20.9	16.6	13.1	11.7	9.3

XL - Nickelplated Steel Round Wound

Item#	Unit Weight	b	a	a	f	e	d	e	B
NW017	.00005524	-	18.0	14.3	11.3	10.1	8.0	6.4	5.7
NW018	.00006215	25.5	20.2	16.1	12.8	11.4	9.0	7.2	6.4
NW019	.00006947	28.5	22.6	18.0	14.3	12.7	10.1	8.0	7.1
NW020	.00007495	-	24.4	19.4	15.4	13.7	10.9	8.6	7.7
NW021	.00008293	34.0	27.0	21.4	17.0	15.2	12.0	9.6	8.5
NW022	.00009184	37.7	29.9	23.7	18.8	16.8	13.3	10.6	9.4
NW024	.00010857	44.6	35.4	28.1	22.3	19.8	15.8	12.5	11.1
NW026	.00012671	-	41.3	32.8	26.0	23.2	18.4	14.6	13.0
Item#	Unit Weight	f <th>e</th> <th>d</th> <th>e</th> <th>B</th> <th>A</th> <th>G</th> <th>F</th>	e	d	e	B	A	G	F
NW028	.00014666	30.1	26.8	21.3	16.9	15.1	11.9	9.5	7.5
NW030	.00017236	35.4	31.5	25.0	19.9	17.7	14.0	11.1	8.8
NW032	.00019347	39.7	35.4	28.1	22.3	19.9	15.8	12.5	9.9
NW034	.00021590	44.3	39.5	31.3	24.9	22.2	17.6	14.0	11.1
NW036	.00023964	49.2	43.8	34.8	27.6	24.6	19.5	15.5	12.3
NW038	.00026471	54.3	48.4	38.4	30.5	27.2	21.6	17.1	13.6
Item#	Unit Weight	c	B	A	G	E	D	C	
NW039	.00027932	32.2	28.7	22.8	18.1	14.3	12.8	10.1	8.0
NW042	.00032279	37.2	33.1	26.3	20.9	16.6	14.8	11.7	9.3
NW044	.00035182	40.5	36.1	28.7	22.7	18.0	16.1	12.8	10.1
NW046	.00038216	44.0	39.2	31.1	24.7	19.6	17.5	13.9	11.0
NW048	.00041382	47.7	42.5	33.7	26.8	21.2	18.9	15.0	11.9
NW049	.00043014	49.5	44.2	35.0	27.8	22.1	19.7	15.6	12.4
NW052	.00048109	55.4	49.4	39.2	31.1	24.7	22.0	17.4	13.9
Item#	Unit Weight	A	G	E	D	C	B'	A'	
NW054	.00053838	43.9	34.8	27.6	24.6	19.5	15.5	13.8	11.0
NW056	.00057598	46.9	37.2	29.5	26.3	20.9	16.6	14.8	11.7
NW059	.00064191	52.3	41.5	32.9	29.3	23.3	18.5	16.4	13.1
NW060	.00066542	54.2	43.0	34.1	30.4	24.1	19.2	17.1	13.5
NW062	.00070697	57.6	45.7	36.3	32.3	25.6	20.4	18.1	14.4
NW064	.00074984	61.1	48.5	38.5	34.3	27.2	21.6	19.2	15.3
NW066	.00079889	65.1	51.6	41.0	36.5	29.0	23.0	20.5	16.3
NW068	.00084614	68.9	54.7	43.4	38.7	30.7	24.4	21.7	17.2
NW070	.00089304	72.7	57.7	45.8	40.8	32.4	25.7	22.9	18.2
NW072	.00094124	76.7	60.8	48.3	43.0	34.1	27.1	24.1	19.2
NW074	.00098869	80.5	63.9	50.7	45.2	35.9	28.5	25.3	20.1
NW080	.0115011	93.7	74.4	59.0	52.6	41.7	33.1	29.5	23.4

XL - Nickelplated Steel Round Wound Steinberger

Item#	Unit Weight	b	a	a	f	e	d	e	B
SNW024	.00010857	44.6	35.4	28.1	22.3	19.8	15.8	12.5	11.1
SNW026	.00012671	-	41.3	32.8	26.0	23.2	18.4	14.6	13.0
Item#	Unit Weight	f	e	d	e	B	A	G	F
SNW032	.00019347	39.7	35.4	28.1	22.3	19.9	15.8	12.5	9.9
SNW036	.00023964	49.2	43.8	34.8	27.6	24.6	19.5	15.5	12.3
Item#	Unit Weight	e	B	A	G	E	D	C	
SNW042	.00032279	37.2	33.1	26.3	20.9	16.6	14.8	11.7	9.3
SNW046	.00038216	44.0	39.2	31.1	24.7	19.6	17.5	13.9	11.0

XLS - Stainless Steel Round Wound

Item#	Unit Weight	b	a	a	f	e	d	e	B
XSG018	.00006588	27.0	21.5	17.0	13.5	12.0	9.6	7.6	6.8
XSG020	.00007919	-	25.8	20.5	16.2	14.5	11.5	9.1	8.1
XSG021	.00008774	-	28.6	22.7	18.0	16.0	12.7	10.1	9.0
XSG022	.00009731	39.9	31.7	25.2	20.0	17.8	14.1	11.2	10.0
XSG024	.00011501	-	37.5	29.7	23.6	21.0	16.7	13.2	11.8
XSG026	.00013419	-	43.7	34.7	27.5	24.5	19.5	15.5	13.8

Figura 3.2 Especificación de la tensión de las cuerdas. [8]

3.1.2 Elección de componentes de la reductora, relación de transmisión, etc...

Debido a la gran tensión de las cuerdas, haría falta un motor muy potente para poder tensar o destensar las cuerdas directamente. Además es necesario que el mecanismo no sea reversible, es decir, que la transmisión de potencia solo pueda darse en una dirección, (el motor hace girar las clavijas en ambos sentidos, horario y antihorario, pero las clavijas no pueden hacer girar el motor), de no ser así, se perdería la afinación. Esto obliga a incorporar al menos una pareja de engranajes corona –tornillo sin fin.

Llegados a este punto se plantean dos opciones:

- a) Diseñar cada pieza del mecanismo, con las parejas de engranajes necesarias para poder disminuir al máximo el tamaño de los motores, mecanizar los ejes y ensamblar todo el conjunto. Esta opción evita las restricciones de tamaño y forma consecuentes de comprar componentes comerciales, pero por otro lado, implica unos costes muy altos y un alto riesgo de que por fallos de diseño o fabricación el mecanismo no funcione correctamente.

Además, es una opción aplicable al desarrollo de un producto comercializable, lo cual no es el objetivo del proyecto. Por lo tanto, desde este punto, se abandona esta opción.

- b) Escoger componentes comerciales y ensamblarlos, comprobando teóricamente que son resistentes a los esfuerzos mecánicos a lo que se les va a someter.

Se pretende construir un prototipo, que si bien no es óptimo (tamaño, forma, acabado, etc.), sea completamente funcional y económicamente viable.

Lo que significa que los componentes se limitan a los existentes en el mercado. La reducción de par se va a llevar a cabo únicamente con una pareja corona-tornillo sinfín.

Como consecuencia, habrá que encontrar un motor que se adapte a estas restricciones de par y potencia impuestas por la tensión de la cuerda y los engranajes.

Este sistema presenta algunas ventajas respecto al primero, los engranajes vienen protegidos y montados en sus ejes con cojinetes y el único problema mecánico a tener en cuenta es el acople de motor y reductora.

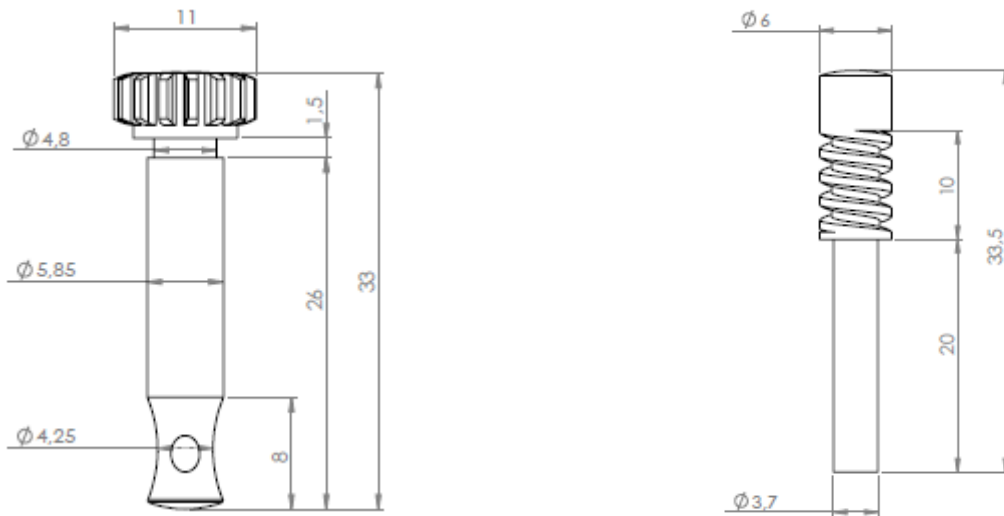
3.1.3 Características geométricas de la reductora comercial.

La reductora es una clavija de guitarra compuesta por una carcasa exterior que protege una pareja corona- tornillo sin fin. El eje del motor se acopla directamente al eje del sin fin de la clavija.



Figura 3.3 Clavija de guitarra [9]

Desmontando la clavija, midiendo los engranajes y haciéndolos en 3D mediante SolidWorks se obtiene:



Corona		
Nº dientes	N_G	15
Diámetro de paso	d_G	9 mm
Ancho efectivo	F_e	3,5 mm

Tornillo sin fin		
Nº hilos	N_W	1
Diámetro de paso	d_W	4,5 mm
Paso axial	p_x	2 mm

El resto de características geométricas hay que calcularlas, la relación de transmisión es:

$$i = \frac{N_G}{N_W} = 15/1$$

El avance:

$$L = p_x \cdot N_W = 2 \text{ mm}$$

El ángulo de avance correspondiente:

$$\lambda = \arctan\left(\frac{L}{\pi \cdot d_W}\right) \approx 8^\circ$$

El ángulo de presión normal se obtiene de la “Figura 3”, ya que el ángulo de avance es menor de 15°:

$$\phi_n = 14,5^\circ$$

Ángulos de presión y profundidades de dientes recomendados para engranajes de tornillo sinfín

Ángulo de avance λ , grados	Ángulo de presión ϕ_n , grados	Cabeza α	Raíz b_c
0-15	$14\frac{1}{2}$	$0.3683p_x$	$0.3683p_x$
15-30	20	$0.3683p_x$	$0.3683p_x$
30-35	25	$0.2865p_x$	$0.3314p_x$
35-40	25	$0.2546p_x$	$0.2947p_x$
40-45	30	$0.2228p_x$	$0.2578p_x$

Figura 3.4 Ángulos de presión y profundidades de dientes para engranajes de tornillo sinfín [10]

3.1.4 Selección del motor

La elección del motor depende de varios factores. Primero, el motor tiene que desarrollar suficiente par como para hacer girar fácilmente la clavija, segundo, tiene que ser potente para mover el mecanismo a velocidades altas, y tercero, debe ser un motor muy preciso.

Estas restricciones, alto par de arranque, potencia y precisión, apuntan hacia un motor paso a paso.

Para estimar el par que tendrá que mover el motor, primero hay que calcular el momento que ejerce la cuerda en la clavija. A la altura donde se enrolla la cuerda el diámetro del eje es 4,5 mm, por lo tanto:

$$M_{cuerda} = T_{cuerda} \cdot r_{clavija} = 195 \text{ mN} \cdot \text{m}$$

A diferencia del resto de engranajes cuya eficiencia es alrededor del 96-98% debido a que el movimiento en el punto de paso es rodamiento puro, con pérdidas casi despreciables, el tornillo sin fin tiene una eficiencia bastante menor (60-80%) debido al rozamiento producido por un movimiento de deslizamiento puro.

El coeficiente de rozamiento del tornillo sinfín depende de la velocidad deslizamiento, que es la suma vectorial de la velocidad de la corona y la del tornillo sinfín en sus respectivos diámetros de paso.

$$\vec{V}_s = \vec{V}_G + \vec{V}_W$$

La velocidad de deslizamiento no es constante, porque según la frecuencia de la cuerda se acerque a la de la nota objetivo, disminuirá la velocidad del motor con el fin de aumentar la precisión. Por lo tanto, para estos cálculos preliminares, se supone un valor típico del coeficiente de rozamiento $f = 0,05$ que se recalculará más adelante. La eficiencia es:

$$e = \frac{\cos(\phi_n) - f \cdot \tan(\lambda)}{\cos(\phi_n) + f \cdot \cot(\lambda)} \approx 72,6\%$$

Ahora se puede calcular el par que desarrolla el motor:

$$M_{motor} = \frac{M_{cuerda}}{e \cdot i} = 17,9 \text{ mN} \cdot \text{m}$$

Para asegurar que el motor va a funcionar correctamente, el par máximo del motor elegido es 40 mN·m y cada paso avanza 1,8°. El resto de características se verán en la parte de electrónica.

3.1.5 Calculo de fuerzas en los engranajes.

La fuerza que la corona ejerce sobre el sinfín se puede descomponer de la siguiente manera:

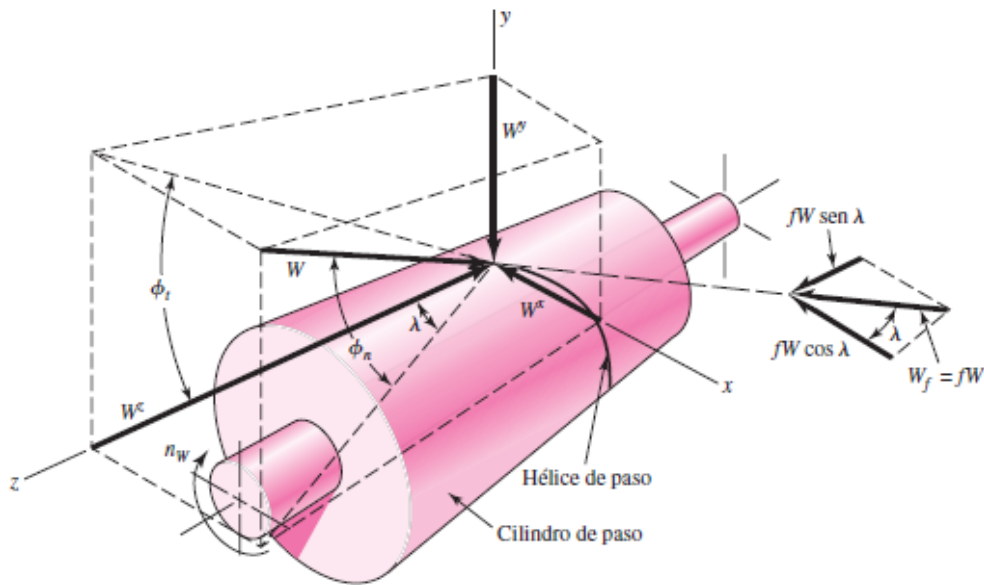


Figura 3.5 Descomposición de fuerzas en un tornillo sinfín [11]

$$W_x = W_{Wt} = -W_{Ga} = W \cdot (\cos(\phi_n) \cdot \text{sen}(\lambda) + f \cdot \cos(\lambda))$$

$$W_y = W_{Wr} = -W_{Gr} = W \cdot \text{sen}(\phi_n)$$

$$W_z = W_{Wa} = -W_{Gt} = W \cdot (\cos(\phi_n) \cdot \cos(\lambda) - f \cdot \text{sen}(\lambda))$$

Donde el subíndice “W” hace referencia al sinfín, “G” a la corona, y los subíndices “a”, “r”, “t”, a las direcciones axial, radial y tangencial respectivamente.

Para calcular la eficiencia real, primero hay que conocer la potencia a la salida del engranaje, en la Figura 3.6 se relaciona el par con el número de pasos por segundo del motor, con ella se puede calcular la velocidad a la salida. Desarrollando un par de 1,7 N·cm = 173,3 gf·cm el motor es capaz de girar a aproximadamente 3300 pasos por segundo. El motor gira 1,8° por paso y la relación de transmisión de los engranajes es 15/1, por lo tanto la velocidad angular del sinfín es:

$$n_W = \frac{3300 \cdot 1,8}{360} = 15,6 \text{ rps} = 990 \text{ rpm}$$

La velocidad angular de la corona es:

$$n_G = \frac{n_W}{i} = 66 \text{ rpm}$$

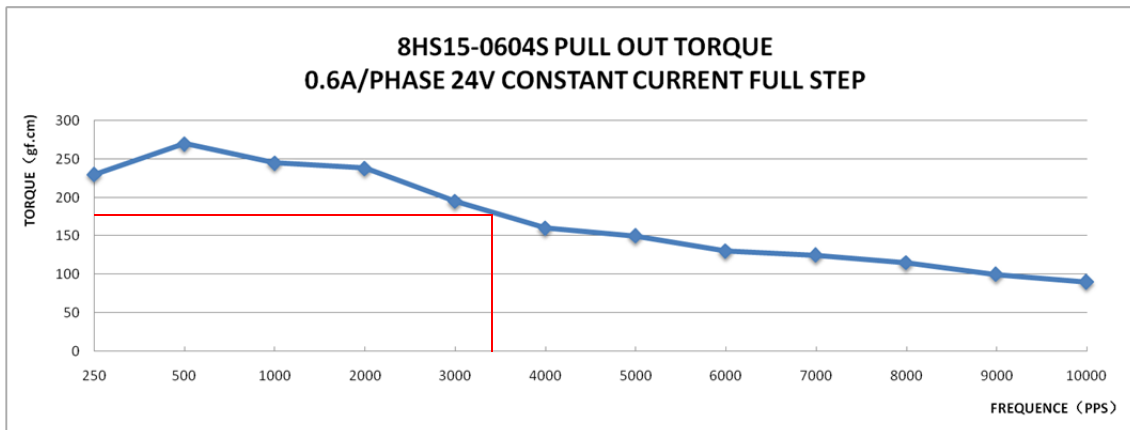


Figura 3.6 Par motor vs frecuencia (pasos por segundo) [12]

Con estas velocidades angulares se calcula la velocidad de deslizamiento, esta velocidad se usa para calcular el coeficiente de rozamiento:

$$V_s = \frac{\pi \cdot d_W \cdot n_W}{60 \cdot \cos(\lambda)} = 0,24 \frac{m}{s}$$

$$V_s = \frac{\pi \cdot d_W \cdot n_W}{12 \cdot \cos(\lambda)} = 46,37 \frac{pies}{min}$$

La AGMA proporciona el coeficiente de fricción como:

$$f = 0,103 \cdot \exp(-0,110 \cdot V_s^{0,450}) + 0,012 = 0,067 \quad V_s > 10 \text{ pies/min}$$

Para calcular la fuerza que aplican los engranajes hay que calcular la velocidad del sinfín en el diámetro de paso:

$$V_W = \frac{\pi \cdot d_W \cdot n_W}{60} = 0,23 \frac{m}{s}$$

Y con la potencia a la entrada se pueden calcular las fuerzas descompuestas. Estas fuerzas se utilizarán para el cálculo de los ejes.

$$W_{W_t} = -W_{Ga} = \frac{2 \cdot \pi \cdot M_{motor} \cdot n_W}{60 \cdot V_W} = 8,07 \text{ N}$$

Ahora se puede calcular la fuerza total y seguidamente el resto de las fuerzas en los ejes:

$$W = \frac{W_{W_t}}{\cos(\phi_n) \cdot \text{sen}(\lambda) + f \cdot \cos(\lambda)} = 40,13 \text{ N}$$

Y el resto de fuerzas:

$$W_{Wr} = -W_{Gr} = W \cdot \text{sen}(\phi_n) = 10,05 \text{ N}$$

$$W_{Wa} = -W_{Gt} = W \cdot (\cos(\phi_n) \cdot \cos(\lambda) - f \cdot \text{sen}(\lambda)) = 38,1 \text{ N}$$

3.1.6 Calculo de la resistencia del engranaje.

Lo primero que se debe comprobar es si el engranaje podrá aguantar el desgaste. La AGMA relaciona fuerza tangencial permisible en el diente del engrane con otros parámetros:

$$W_{tperm} = C_s \cdot D_m^{0,8} \cdot F_e \cdot C_m \cdot C_v \quad [13]$$

Las ecuaciones están definidas por la AGMA, por lo que el sistema de unidades es el imperial, y las unidades de la fuerza máxima permisible serán lbf (libras de fuerza) que después habrá que convertir al SI (Newtons).

Al ser la fuerza tangencial de la corona mayor que la del sinfín, es la más restrictiva, por lo que los cálculos se hacen para la corona.

- C_s es el factor de materiales, se calcula con la distancia entre centros (C) de los engranajes en pulgadas:

$$C = \frac{d_G + d_W}{2} = 6,75 \text{ mm} = 0,266 \text{ pulg}$$

$$C_s = 270 + 10,37 \cdot C^3 = 270,19$$

- C_m es el factor de corrección de relación y depende de la relación de transmisión (i):

$$C_m = 0,02 \cdot \sqrt{-i^2 + 40 \cdot i - 76} + 0,46 = 0,805$$

- C_v es el factor de velocidad se calcula con la velocidad de deslizamiento (V_s) en pies/minuto:

$$C_v = 0,659 \cdot \exp(-0,0011 \cdot V_s) = 0,626$$

Por lo tanto la fuerza máxima permisible es:

$$W_{tperm} = C_s \cdot \left(\frac{d_G}{25,4}\right)^{0,8} \cdot \frac{F_e}{25,4} \cdot C_m \cdot C_v = 10,17 \text{ lbf} = 45,25 \text{ N} > W_{Gt}$$

La fuerza tangencial máxima de la corona es menor que la fuerza permisible, así que el engranaje es válido para cualquier régimen de trabajo.

Hay que comprobar también el esfuerzo de flexión en un diente del engranaje, esto se hace con la ecuación de Lewis adaptada por Buckingham. Como los dientes del sinfín son intrínsecamente más fuertes que los de la corona, no se consideran para la comprobación.

$$\sigma_a = \frac{W_{Gt}}{p_n \cdot F_e \cdot y} = 12,10 \text{ MPa}$$

Donde $p_n = p_x \cdot \cos \lambda$, e "y" es el factor de forma de Lewis, que para $\phi_n = 14,5^\circ$ es $y = 0,1$.

El límite elástico del bronce es aproximadamente de 70 MPa (anexo A), por lo tanto, el engranaje es válido.

3.1.7 Calculo de fuerzas en los ejes.

Para comprobar si los ejes aguantarán los esfuerzos a los que van a estar sometidos, hay que calcular las reacciones en los apoyos a las fuerzas que se transmiten desde el motor a la cuerda. En este sistema existen únicamente dos ejes, el eje motor y el eje de la clavija.

Para simplificar la notación, las fuerzas en el tornillo sinfín se van a llamar W y las de la corona G, los subíndices a, r, t indican las direcciones axial, radial y tangencial respectivamente. Las reacciones en los apoyos A y B son F_a y F_b y sus componentes en las tres direcciones son X, Y y Z con los subíndices a y b para indicar a que apoyo pertenece cada fuerza. Además, i, j, k son vectores unitarios en las direcciones de los ejes.

Se ha despreciado la masa de los ejes y engranajes. La fuerza de rozamiento, pese a que no aparezca, se ha tenido en cuenta en el cálculo de las fuerzas de los engranajes.

3.1.7.1 Calculo de fuerzas para el eje del sinfín:

El eje motor tiene en un extremo el motor y en el otro el tornillo sinfín y los apoyos se sitúan a ambos lados del sinfín.

Ecuaciones de equilibrio de fuerzas para el **eje del sinfín**:

$$\sum M_A = \overline{M}_m + \vec{D}_{AC} \cdot \vec{W} + \vec{D}_{AB} \cdot \vec{F}_B = 0$$

$$\sum F = \vec{F}_A + \vec{F}_B + \vec{W} = 0$$

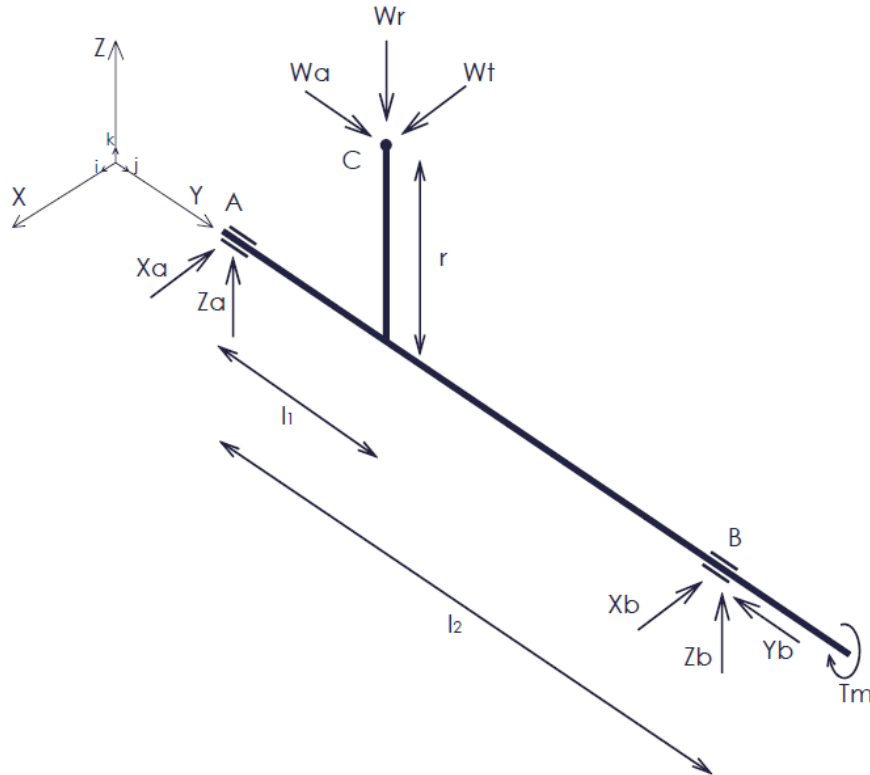


Figura 3.7 Esquema de fuerzas en el eje del sinfín

Donde:

$$\overline{M}_m = M_{motor} \cdot (-j) = 0,0179 \cdot (-j) \text{ N} \cdot \text{m}$$

$$\overline{W} = W_t \cdot (i) + W_a \cdot (j) + W_r \cdot (-z) = 8,07 \cdot (i) + 38,1 \cdot (j) + 10,05 \cdot (-z) \text{ N}$$

$$\overline{F}_A = X_A(-i) + Z_A(-k)$$

$$\overline{F}_B = X_B(-i) + Y_B(-j) + Z_B(-k)$$

$$\overline{D}_{AC} = l_1 \cdot (j) + r \cdot (k) = 0,007 \cdot (j) + 0,0023 \cdot (k) \text{ m}$$

$$\overline{D}_{AB} = l_2 \cdot (j) = 0,014 \cdot (j)$$

Sustituyendo y operando en el sumatorio de momentos:

$$\sum M_A = (Z_B \cdot l_2 - W_a \cdot r - W_r \cdot l_1) \cdot i + (W_t \cdot r - M_{motor}) \cdot j + (X_B \cdot l_2 - W_t \cdot l_1) \cdot z = 0$$

Por lo tanto:

$$Z_B = \frac{W_a \cdot r + W_r \cdot l_1}{l_2} = 11,14 \text{ N}$$

$$X_B = \frac{W_t \cdot l_1}{l_2} = 4,04 \text{ N}$$

Haciendo equilibrio de fuerzas:

$$\sum F = (W_t - X_A - X_B) \cdot i + (W_a - Y_B) \cdot j + (Z_A + Z_B - W_r) \cdot k = 0$$

Sustituyendo los valores de Z_B y X_B se pueden calcular el resto de reacciones:

$$X_A = W_t \cdot \left(1 - \frac{l_1}{l_2}\right) = 4,04 \text{ N}$$

$$Y_B = W_a = 38,1 \text{ N}$$

$$Z_A = W_r \cdot \frac{(l_2 - l_1)}{l_2} - W_a \cdot \frac{r}{l_2} = -1,1 \text{ N}$$

3.1.7.2 Calculo de fuerzas para el eje de la corona:

El eje de la corona tiene en un extremo un taladro pasante para sujetar la cuerda de la guitarra y en el otro la corona del sinfín. Este eje, está sujeto por un cojinete de fricción en la parte central, con una pequeña holgura entre ambos para que el eje pueda girar. Al aplicar las fuerzas del engranaje y de la cuerda sobre el eje, las reacciones sobre el eje solo van a existir en los extremos del cojinete. Se puede ver en la Figura 3.8: Cuando no existen fuerzas el eje tiene un apoyo continuo sobre el cojinete. Sin embargo, al aplicarle las fuerzas, se desplaza como un mecanismo hasta que queda apoyado únicamente sobre los puntos A y B.

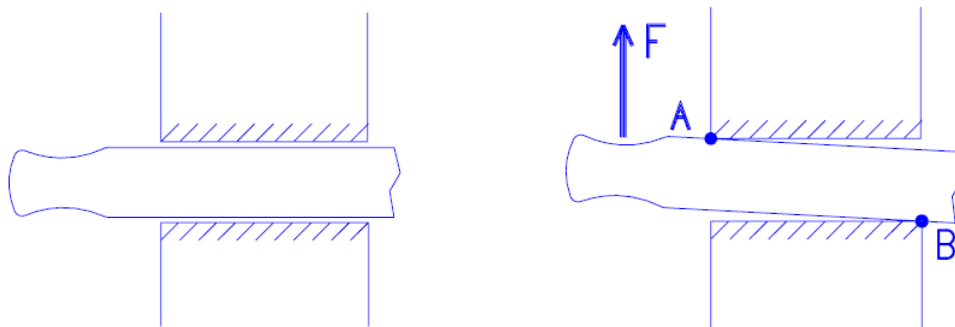


Figura 3.8 Explicación de los apoyos del eje de la corona

Se hace el equilibrio de fuerzas y momentos en el eje de la corona:

$$\sum M_D = \vec{D}_{DG} \cdot \vec{T}_c + \vec{D}_{DF} \cdot \vec{G} + \vec{D}_{DE} \cdot \vec{F}_E + \vec{M}_c = 0$$

$$\sum F = \vec{F}_D + \vec{F}_E + \vec{G} = 0$$

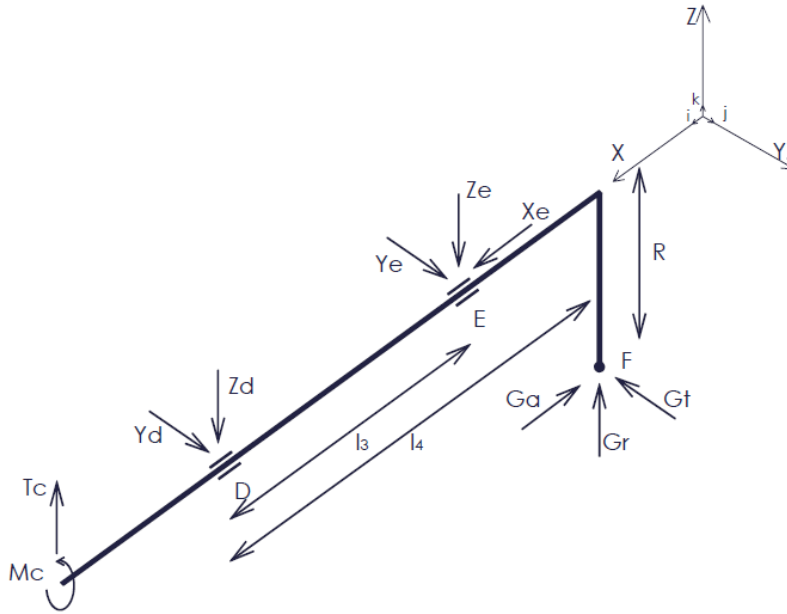


Figura 3.9 Esquema de fuerzas en el eje de la corona.

Donde:

$$\vec{M}_C = M_{\text{cuerda}} \cdot (i) = 0,195 \cdot (i) \text{ N} \cdot \text{m}$$

$$\vec{T}_C = T_{\text{cuerda}} \cdot (k) = 86,8 \cdot (k) \text{ N}$$

$$\vec{G} = G_a \cdot (-i) + G_t \cdot (-j) + G_r \cdot (z) = 8,07 \cdot (i) + 38,1 \cdot (j) + 10,05 \cdot (-z) \text{ N}$$

$$\vec{F}_D = Y_D(j) + Z_D(-k)$$

$$\vec{F}_E = X_E(i) + Y_E(j) + Z_E(-k)$$

$$\vec{D}_{DE} = l_3 \cdot (-i) = 0,018 \cdot (-i) \text{ m}$$

$$\vec{D}_{DF} = l_4 \cdot (-i) + R \cdot (-k) = 0,023 \cdot (-i) + 0,0045 \cdot (-k) \text{ m}$$

$$\vec{D}_{DG} = l_5 \cdot (i) = 0,07 \cdot (i)$$

Sustituyendo y operando en el sumatorio de momentos:

$$\sum M_D = (M_c - G_t \cdot R) \cdot i + (G_r \cdot l_4 + G_a \cdot R - Z_E \cdot l_3 - T_c \cdot l_5) \cdot j + (G_t \cdot l_4 - Y_E \cdot l_3) \cdot k = 0$$

Se pueden despejar Z_E e Y_E :

$$Y_E = \frac{G_t \cdot l_4}{l_3} = 48,68 \text{ N}$$

$$Z_E = \frac{G_r \cdot l_4 + G_a \cdot R - T_c \cdot l_5}{l_3} = -18,9 \text{ N}$$

Si se resuelve el equilibrio de fuerzas:

$$\sum F = (X_E - G_a) \cdot i + (Y_D + Y_E - G_t) \cdot j + (G_r + T_c - Z_D - Z_E) \cdot k = 0$$

Se pueden calcular el resto de reacciones:

$$X_E = G_a = 8,07 \text{ N}$$

$$Y_D = G_t - Y_E = -10,58 \text{ N}$$

$$Z_D = G_r + T_c - Z_E = 115,75 \text{ N}$$

Una vez que están calculadas todas las reacciones en los apoyos se puede calcular el diagrama de momentos flectores, axiles y cortantes para cada eje. Una vez calculados, se buscará el punto más crítico del eje para comprobar teóricamente si aguantara los esfuerzos a los que va a estar sometido.

3.1.8 Diagrama de momentos axiles, flectores y torsores.

El **eje del sinfín** tiene las siguientes leyes de momentos:

Momentos en X: (el sentido antihorario se considera positivo)

- Para $0 < a < l_1 \rightarrow M_x = Z_A \cdot a$
- Para $l_1 < a < l_2 \rightarrow M_x = Z_A \cdot a + W_a \cdot r - W_r \cdot (a - l_1)$

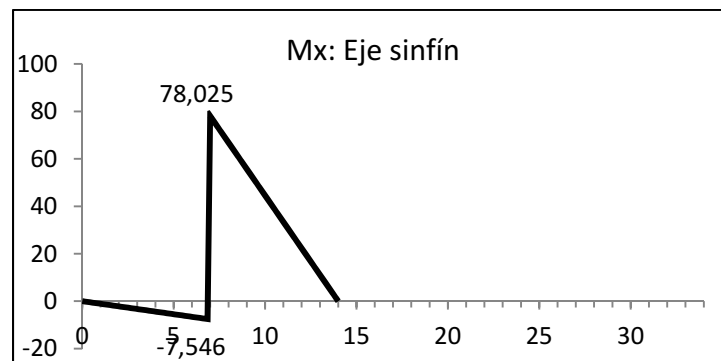


Figura 3.10 Diagrama de flectores en el eje X.

Momentos en Y: (torsor)

- Para $0 < a < l_1 \rightarrow M_y = 0$
- Para $l_1 < a < l_{eje \ sinfín} \rightarrow M_z = W_t \cdot r$

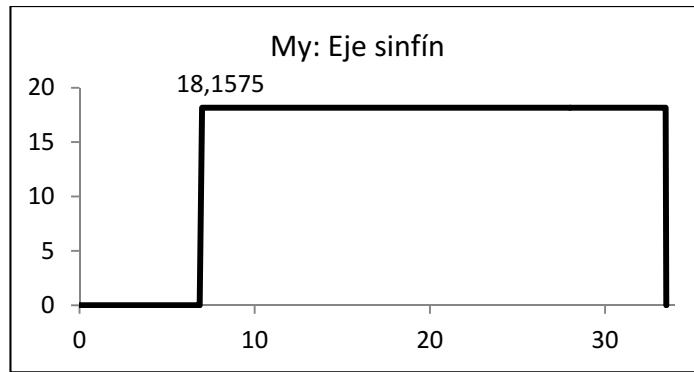


Figura 3.11 Diagrama de torsores en el eje Y

Momentos en Z: (el sentido antihorario se considera positivo)

- Para $0 < a < l_1 \rightarrow M_z = X_A \cdot a$
- Para $l_1 < a < l_2 \rightarrow M_z = X_A \cdot a - W_t \cdot (a - l_1)$

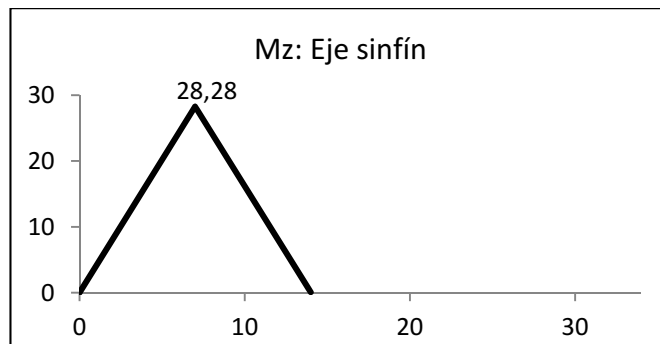


Figura 3.12 Diagrama de flectores en el eje Z.

Axiles en Y: (compresión positivo)

- Para $0 < a < l_1 \rightarrow N_y = 0$
- Para $l_1 < a < l_2 \rightarrow N_y = W_a$

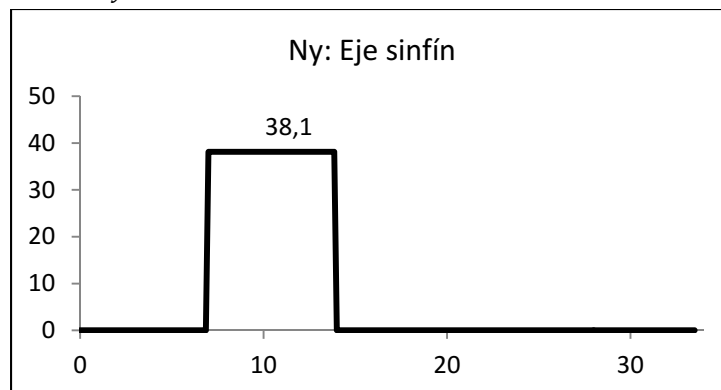


Figura 3.13 Diagrama de axiles en el eje Y.

De la misma manera se calculan las leyes de momentos del **eje de la corona**:

Momentos en X: (torsor)

- Para $0 < a < l_{\text{eje corona}} \rightarrow M_x = M_c$

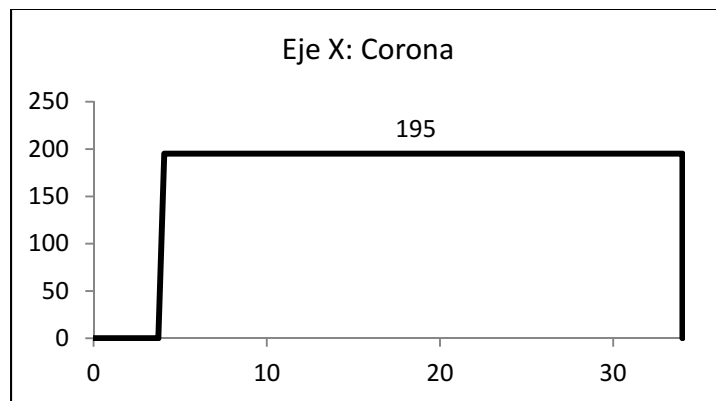


Figura 3.14 Diagrama de torsores en el eje X

Momentos en Y: (el sentido antihorario se considera positivo)

- Para $0 < a < l_5 \rightarrow M_y = T_c \cdot a$
- Para $l_5 < a < (l_5 + l_3) \rightarrow M_y = T_c \cdot a - Z_D \cdot (a - l_5)$
- Para $(l_5 + l_3) < a < (l_5 + l_4) \rightarrow M_y = T_c \cdot a - Z_D \cdot (a - l_5) - Z_E \cdot (a - l_3 - l_5)$

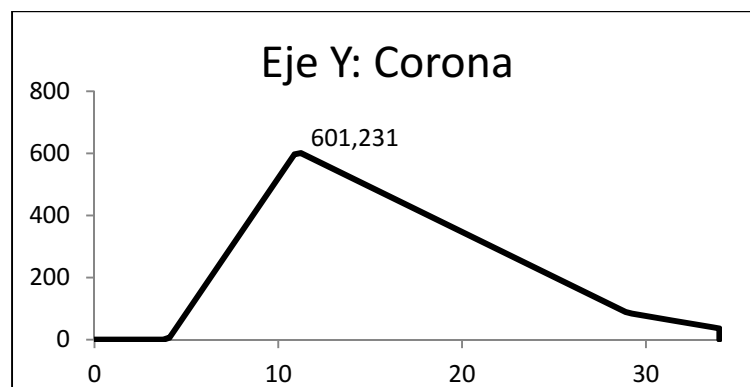


Figura 3.15 Diagrama de flectores en el eje Z.

Momentos en Z: (el sentido antihorario se considera positivo)

- Para $0 < a < l_5 \rightarrow M_z = 0$
- Para $l_5 < a < (l_5 + l_3) \rightarrow M_z = -Y_D \cdot (a - l_5)$
- Para $(l_5 + l_3) < a < (l_5 + l_4) \rightarrow M_z = -Y_D \cdot (a - l_5) - Y_E \cdot (a - l_5 - l_3)$

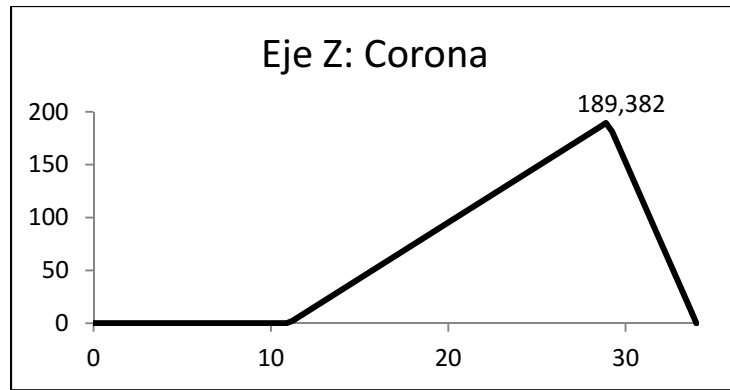


Figura 3.16 Diagrama de flectores en el eje Z.

Axiles en X: (positivo tracción)

- Para $0 < a < l_5 + l_3 \rightarrow N_z = 0$
- Para $l_5 + l_3 < a < l_5 + l_4 \rightarrow N_z = X_E$

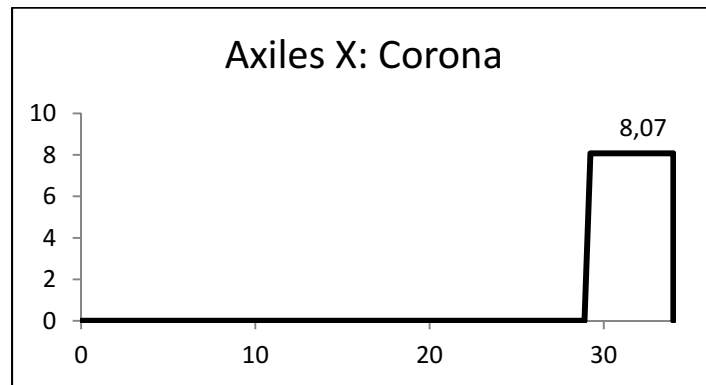


Figura 3.17 Diagrama de axiles en el eje X.

3.1.9 Calculo de la resistencia a fatiga de los ejes

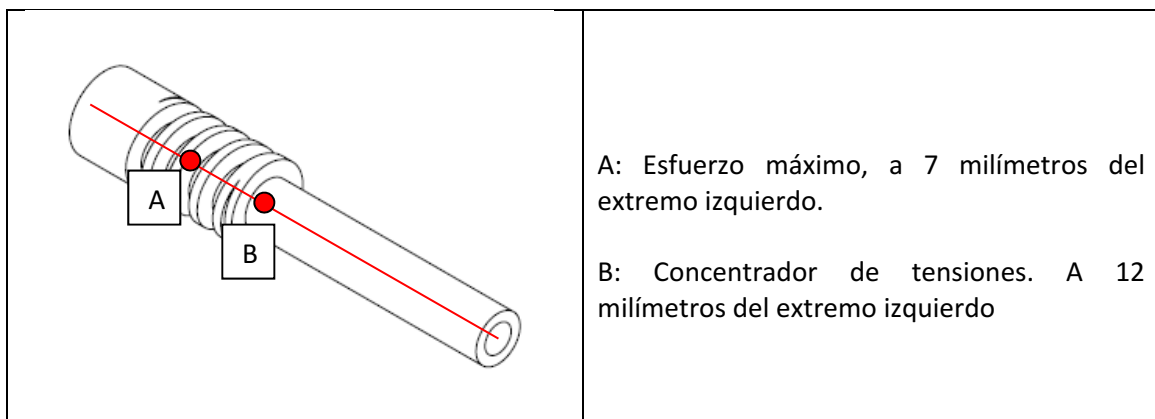
Los esfuerzos se evalúan únicamente en las ubicaciones críticas del eje. Estas ubicaciones se localizan generalmente en la superficie de exterior del eje, cuando está presente el momento torsor y existen concentradores de tensiones.

Lo primero que se tiene que hacer es determinar el punto crítico de cada eje, el criterio a seguir es buscar en las gráficas de momentos flectores y torsores los puntos máximos, y después, observar la geometría del eje en busca de un concentrador de tensión en los puntos cercanos a ese máximo.

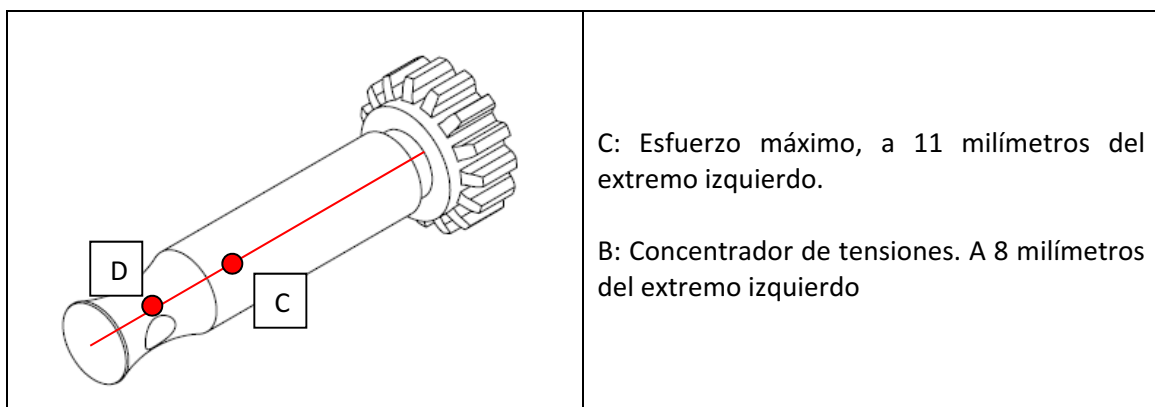
En el eje del sinfín, se puede ver claramente que un posible punto crítico se encuentra justo en el punto de engrane con la corona. Donde termina el sinfín, hay un cambio de diámetro del eje. Como está cerca del punto de máximo esfuerzo, hay que estudiarlo como posible punto crítico. En el eje de la corona, el máximo se encuentra en el apoyo situado más lejos de la corona, también existe un concentrador de tensión en los puntos de alrededor donde el eje se estrecha para que la cuerda pueda dar vueltas.

Puntos de estudio:

Eje sinfín:



Eje corona:



Con las gráficas, se puede calcular el factor concentrador de tensiones de los puntos B y D. “B” (rojo) tiene una relación $D/d=6/3,7$ y $r/d=0,02$. “D” (azul) tiene una relación $D/d=5,85/4,25$ y $r/d>0,3$.

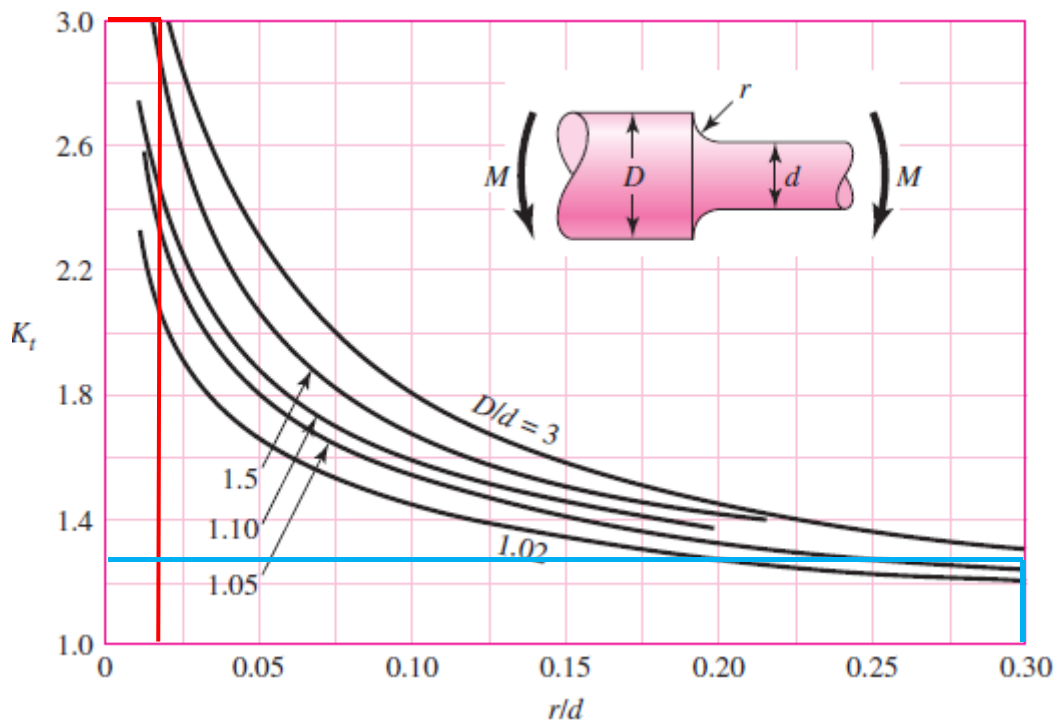


Figura 3.18 Factor teórico de concentración de esfuerzos para ejes sometidos a flexión. [14]

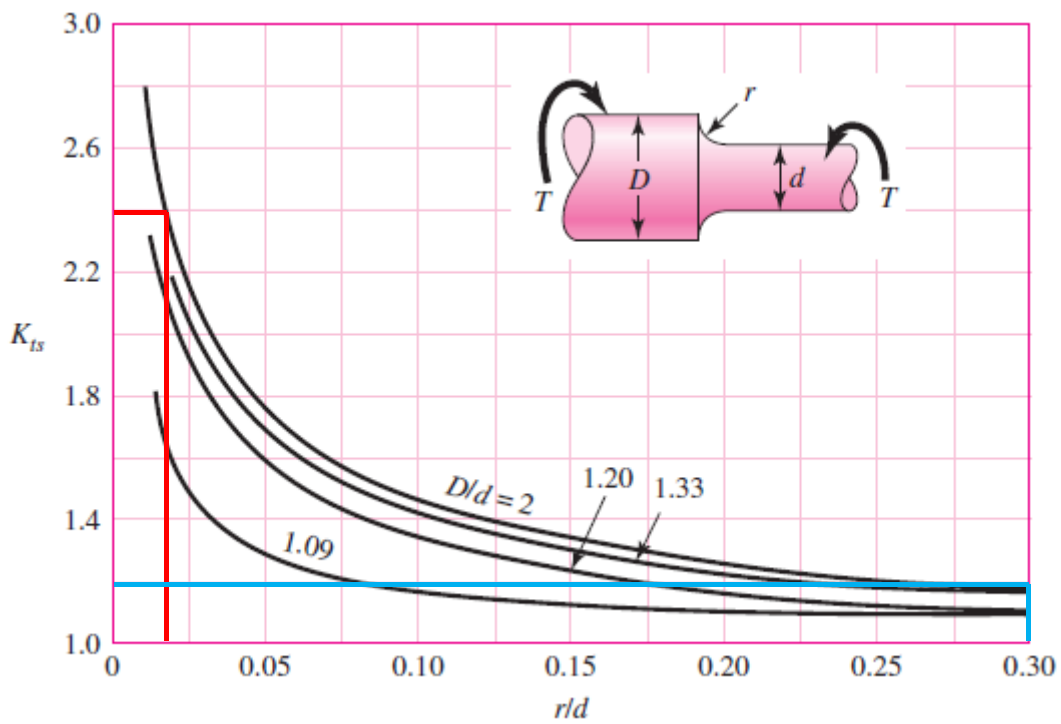


Figura 3.19 Factor teórico de concentración de esfuerzos para ejes sometidos a Torsión. [14]

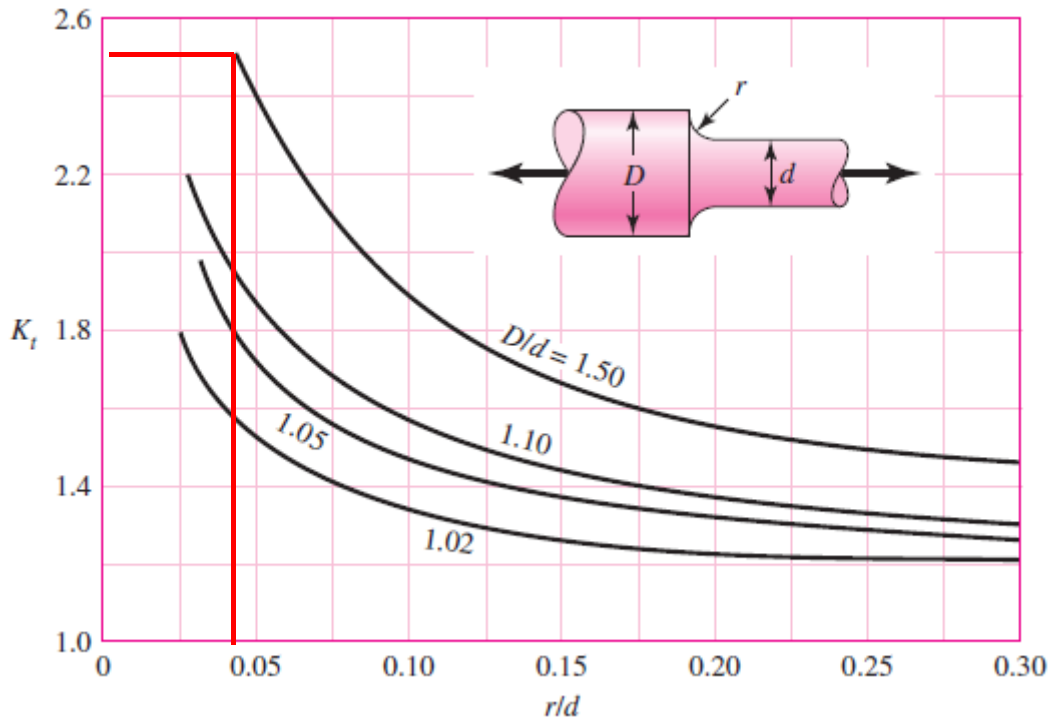


Figura 3.20 Factor teórico de concentración de esfuerzos para ejes sometidos a tracción. [14]

Los ejes son de acero inoxidable, su resistencia última S_{ut} ronda los 0,6 GPa (Anexo A), pero al no saber con seguridad el tipo de acero inoxidable, se va a escoger el mínimo de todos los que hay.

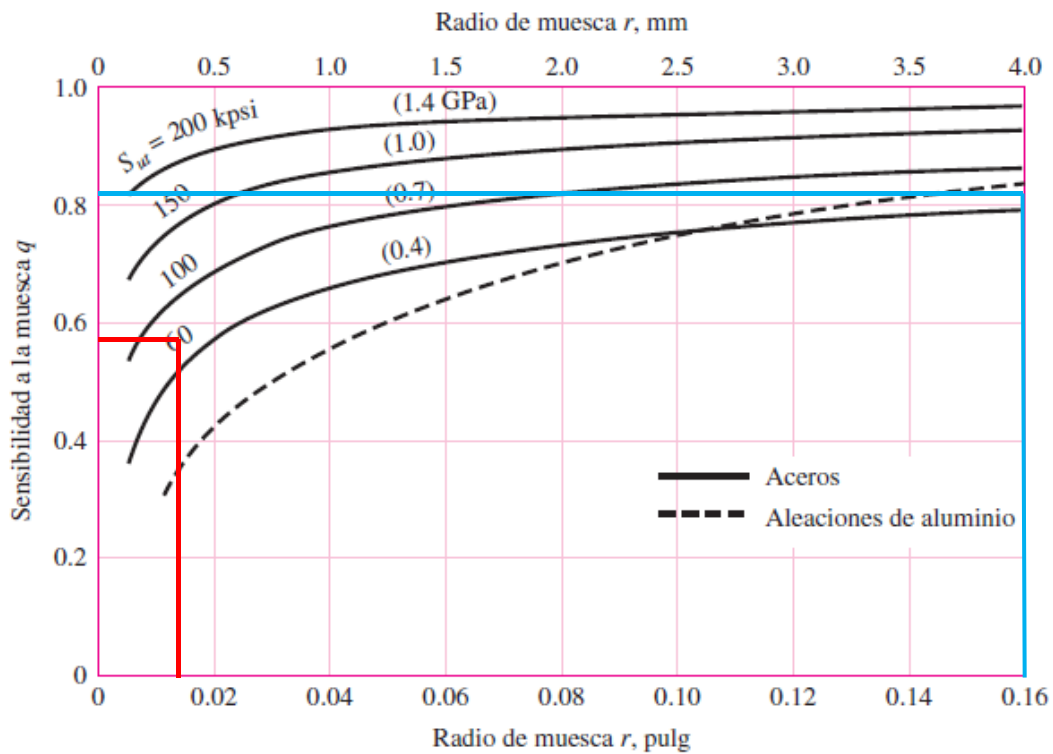


Figura 3.21 Sensibilidad a la muesca en aceros y aleaciones de aluminio sometidos a flexión [15]

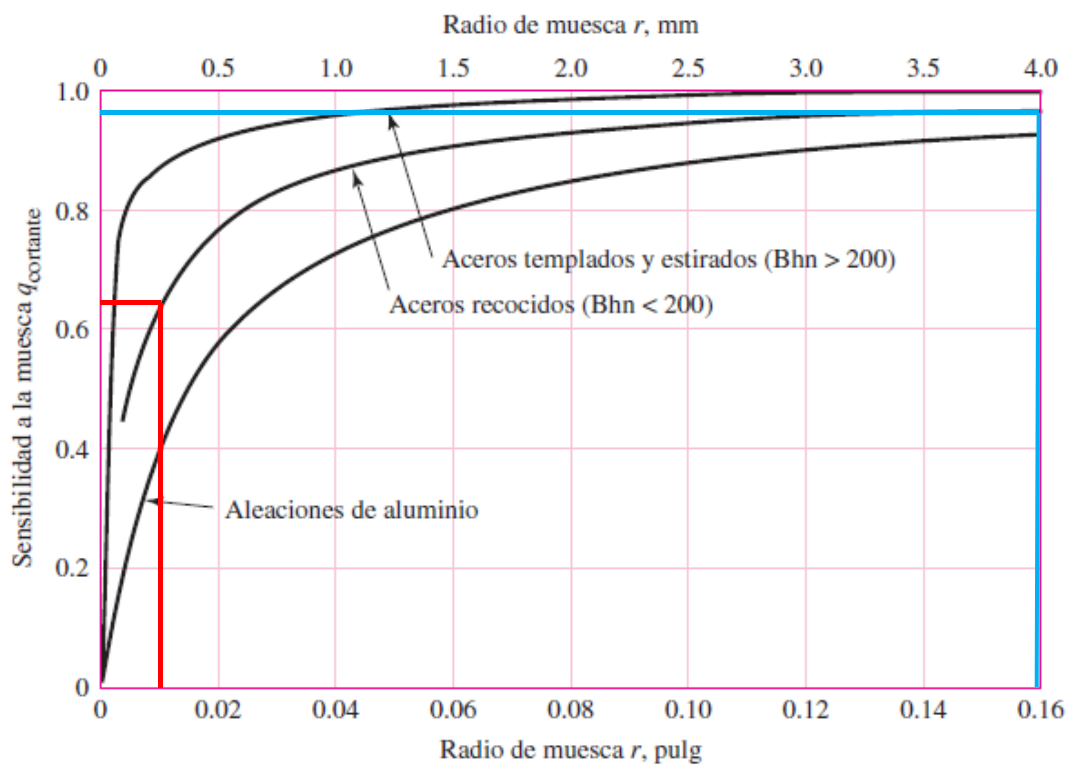


Figura 3.22 Sensibilidad a la muesca de materiales sometidos a torsión inversa [16]

Con los factores teóricos de concentración de esfuerzos K_t y la sensibilidad a la muesca q se puede calcular los factores concentradores de esfuerzos K_f, K_{fs} .

$$K_f = 1 + q \cdot (K_t - 1)$$

$$K_{fs} = 1 + q_{cortante} \cdot (K_{ts} - 1)$$

Para los dos puntos los concentradores de esfuerzos quedan:

Punto B:

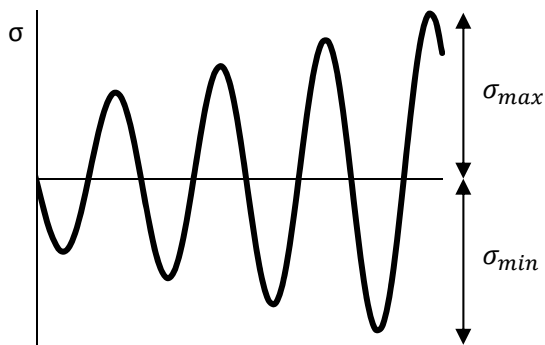
$$\begin{aligned} K_{f,flex} &= 1 + 0,58(3 - 1) = 2,16 \\ K_{fs} &= 1 + 0,64 \cdot (2,4 - 1) = 1,9 \\ K_{f,axial} &= 1 + 0,58(2,5 - 1) = 1,87 \end{aligned}$$

Punto D:

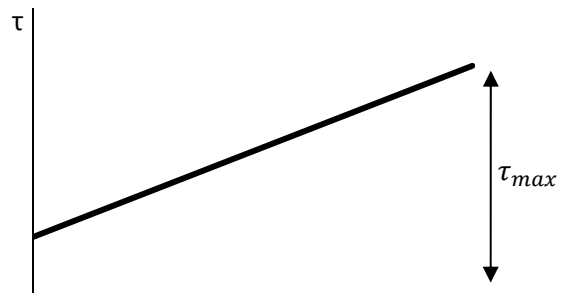
$$\begin{aligned} K_{f,flex} &= 1 + 0,82(1,25 - 1) = 1,2 \\ K_{fs} &= 1 + 0,97 \cdot (1,2 - 1) = 1,19 \end{aligned}$$

Los esfuerzos a los que está sometido un eje que gira son varían a lo largo del tiempo, suelen tener forma sinusoidal los esfuerzos de flexión, en dos ejes y ser constantes los de torsión. En este caso, los esfuerzos de flexión y torsión (en valor absoluto) aumentan y disminuyen conforme se afina la guitarra. En las gráficas se puede observar un ejemplo de la evolución de los esfuerzos con el tiempo.

Esfuerzos de flexión



Esfuerzos de torsión y axiales



Para simplificar los cálculos se puede hacer una suposición conservadora. Los esfuerzos de tracción, flexión y torsión se toman constantes e iguales al esfuerzo máximo ($\sigma_{max} = -\sigma_{min}$, $\tau_{max} = \tau_{min}$), y se pueden simplificar las expresiones igualando σ_m y τ_a a cero. Así, los esfuerzos medios y alternantes serían:

$$\begin{aligned} \sigma_{m,flex} &= \frac{\sigma_{max} + \sigma_{min}}{2} = 0 & \sigma_{a,flex} &= \frac{\sigma_{max} - \sigma_{min}}{2} = \sigma_{max} \\ \sigma_{m,ax} &= \frac{\sigma_{max} + \sigma_{min}}{2} = \sigma_{max} & \sigma_{a,ax} &= \frac{\sigma_{max} - \sigma_{min}}{2} = 0 \\ \tau_m &= \frac{\tau_{max} + \tau_{min}}{2} = \tau_{max} & \tau_a &= \frac{\tau_{max} - \tau_{min}}{2} = 0 \end{aligned}$$

Los ejes son sólidos de sección transversal redonda, así que las expresiones de los esfuerzos medios y alternantes debidos a axiales, flexión y torsión son:

$$\sigma_{a,flex} = K_f \cdot \frac{32 \cdot M_a}{\pi \cdot d^3} \quad \sigma_{m,axial} = K_f \cdot \frac{4 \cdot N_a}{\pi \cdot d^2} \quad \tau_m = K_{fs} \cdot \frac{16 \cdot T_m}{\pi \cdot d^3}$$

Hay que calcular para cada uno de los 4 puntos σ_a y τ_m . Los momentos flectores en distintos planos se deben combinar para obtener momentos totales:

$$M_{total} = \sqrt{M_x^2 + M_y^2}$$

Así se puede calcular M_a

PUNTO A: $K_{f,flex} = 1$, $K_{f,axial} = 1$, $K_{fs} = 1$, $d = 3,7 \text{ mm}$

$$M_a = \sqrt{M_x(7^+)^2 + M_z(7)^2} = 83 \text{ Nmm}$$

$$T_m = 18,175 \text{ Nmm}$$

$$N_m = 38,1$$

$$\sigma_{a,flex} = K_f \cdot \frac{32 \cdot M_a}{\pi \cdot d^3} = 16,69 \text{ MPa}$$

$$\tau_m = K_{fs} \cdot \frac{16 T_m}{\pi \cdot d^3} = 1,82 \text{ MPa}$$

$$\sigma_{m,axial} = K_f \cdot \frac{4 \cdot N_a}{\pi \cdot d^2} = 3,54 \text{ MPa}$$

PUNTO B: $K_{f,flex}=2,16$, $K_{f,axial} = 1$, $K_{fs} = 1,9$, $d = 3,7$

$$M_a = \sqrt{M_X(12)^2 + M_Z(12)^2} = 23,2 \text{ Nmm}$$

$$\sigma_{a,flex} = 10,08 \text{ Mpa}$$

$$T_m = 18,175 \text{ Nmm}$$

$$\tau_m = 3,47 \text{ MPa}$$

$$N_m = 38,1 \text{ N}$$

$$\sigma_{m,axial} = 6,61 \text{ MPa}$$

PUNTO C: $K_f = 1$, $K_{fs} = 1$, $d = 5,85$

$$M_a = \sqrt{M_Y(11)^2 + M_Z(11)^2} = 601,2 \text{ Nmm}$$

$$\sigma_{a,flex} = 30,58 \text{ MPa}$$

$$T_m = 195 \text{ Nmm}$$

$$\tau_m = 4,96 \text{ MPa}$$

$$N_m = 0 \text{ N}$$

PUNTO D: $K_f = 1,2$, $K_{fs} = 1,19$, $d = 4,5$

$$M_a = \sqrt{M_Y(8)^2 + M_Z(8)^2} = 346 \text{ Nmm}$$

$$\sigma_{a,flex} = 46,41 \text{ MPa}$$

$$T_m = 195 \text{ Nmm}$$

$$\tau_m = 12,97 \text{ MPa}$$

$$N_m = 0 \text{ N}$$

Para el cálculo de la resistencia a fatiga, se va a utilizar el criterio de Goodman, que pese a ser conservador, es lineal y por lo tanto sencillo de resolver.

$$\frac{\sigma'_a}{S_e} + \frac{\sigma'_m}{S_{ut}} = \frac{1}{n}$$

El valor n es el coeficiente de seguridad y S_{ut} la resistencia ultima del material

Los valores σ'_m y σ'_a son la combinación de los esfuerzos (flectores y torsores) según la teoría de falla por energía de distorsión para ejes giratorios, redondos y sólidos. Se calculan de la siguiente manera:

$$\sigma'_a = \left[\left(\sigma_{a,flex} + \frac{\sigma_{a,axial}}{0,85} \right)^2 + 3\tau_a^2 \right]^{\frac{1}{2}} = \sigma_{a,flex}$$

$$\sigma'_m = \left[(\sigma_{m.flex} + \sigma_{m.axial})^2 + 3\tau_m^2 \right]^{\frac{1}{2}} = \left[(\sigma_{m.axial})^2 + 3\tau_m^2 \right]^{\frac{1}{2}}$$

Por último antes de hacer la comprobación, falta por calcular S_e , el límite de resistencia a fatiga, con la ecuación de Marín, que mediante la multiplicación de coeficientes, que describen el acabado superficial, la temperatura el tamaño, la carga, consigue extrapolar el límite de fatiga del ensayo de viga rotatoria, S_e' a cualquier elemento mecánico. La ecuación de Marín es:

$$S_e = k_a k_b k_c k_d k_e k_f S_e'$$

Factor de superficie k_a : depende de la calidad de acabado de la superficie y de la resistencia a la tensión:

$$k_a = a \cdot S_{ut}^b$$

Acabado superficial	Factor a		Exponente b
	S_{ut} kpsi	S_{ut} MPa	
Esmerilado	1.34	1.58	-0.085
Maquinado o laminado en frío	2.70	4.51	-0.265
Laminado en caliente	14.4	57.7	-0.718
Como sale de la forja	39.9	272.	-0.995

Figura 3.23 Parámetros en el factor de condición superficial de Marín para aceros [17]

Los dos ejes tienen acabado esmerilado:

$$k_a = 1,58 \cdot 520^{-0,085} = 0,93$$

Factor de tamaño k_b : Para flexión y torsión:

$$k_b = \left(\frac{d}{7,62} \right)^{-0,107} \quad \text{si } 2,79 \leq d < 51 \text{ mm}$$

Puntos A y B: $k_b = 1,08$

Punto C: $k_b = 1,03$

Punto D: $k_b = 1,05$

Factor de temperatura k_c :

$$k_c = 1 \text{ si existen cargas de flexion}$$

Factor de temperatura k_d : cuando las temperaturas de operación son mayores o menores que la temperatura ambiente.

$$k_d = \frac{S_T}{S_{RT}}$$

Temperatura, °C	S_T/S_{RT}	Temperatura, °F	S_T/S_{RT}
20	1.000	70	1.000
50	1.010	100	1.008
100	1.020	200	1.020
150	1.025	300	1.024
200	1.020	400	1.018
250	1.000	500	0.995
300	0.975	600	0.963
350	0.943	700	0.927
400	0.900	800	0.872
450	0.843	900	0.797
500	0.768	1 000	0.698
550	0.672	1 100	0.567
600	0.549		

Figura 3.24 Efecto de la temperatura en la resistencia a la tensión del acero [18]

S_T es la resistencia a la tensión a la temperatura de operación para el acero, y S_{RT} a temperatura ambiente.

Para ambos ejes $k_d = 1$

Factor de confiabilidad k_e :

$$k_e = 1 - 0,08z_a$$

Se toma una confiabilidad del 99%, por lo tanto:

$$k_e = 0,814$$

Confiabilidad, %	Variación de transformación z_a	Factor de confiabilidad k_e
50	0	1.000
90	1.288	0.897
95	1.645	0.868
99	2.326	0.814
99.9	3.091	0.753
99.99	3.719	0.702
99.999	4.265	0.659
99.9999	4.753	0.620

Figura 3.25 Factores de confiabilidad para 8 desviaciones típicas porcentuales [19]

Factor de efectos varios k_f : tiene el propósito de contabilizar la reducción del límite de resistencia a fatiga debida a todos los demás efectos. El problema es que no todos son contabilizables. Se toma para los dos ejes:

$$k_f = 1$$

Límite de resistencia a fatiga, S_e'

$$S_e' = 0,5 S_{ut} \text{ si } S_{ut} < 1400 \text{ MPa}$$

El límite de resistencia a fatiga para los dos ejes es: $S_e' = 260 \text{ MPa}$

Ahora ya se puede aplicar el criterio de Goodman para cada punto:

PUNTO A:

$$S_e = k_a k_b k_c k_d k_e k_f S_e' = 212,57 \text{ MPa}$$

$$\frac{\sigma_{a,flex}}{S_e} + \frac{\sqrt{\sigma_{m,axial}^2 + 3\tau_m^2}}{S_{ut}} = 0,096 \text{ MPa} \rightarrow n = 10$$

PUNTO B:

$$S_e = k_a k_b k_c k_d k_e k_f S_e' = 202,73$$

$$\frac{\sigma_{a,flex}}{S_e} + \frac{\sqrt{\sigma_{m,axial}^2 + 3\tau_m^2}}{S_{ut}} = 0,064 \rightarrow n = 15$$

PUNTO C:

$$S_e = k_a k_b k_c k_d k_e k_f S_e' = 202,73 \text{ MPa}$$

$$\frac{\sigma_{a,flex}}{S_e} + \frac{\sqrt{3\tau_m^2}}{S_{ut}} = 0,167 \text{ MPa} \rightarrow n = 6$$

PUNTO D:

$$S_e = k_a k_b k_c k_d k_e k_f S_e' = 206,66 \text{ MPa}$$

$$\frac{\sigma_{a,flex}}{S_e} + \frac{\sqrt{3\tau_m^2}}{S_{ut}} = 0,268 \text{ MPa} \rightarrow n = 3$$

En vista de que todos los coeficientes de seguridad son mayores que uno, se puede afirmar que ambos ejes aguantan los esfuerzos de fatiga a los que se les va a someter.

3.1.10 Calculo de la resistencia a fluencia.

Hay que contemplar la posibilidad de falla estática en el primer ciclo. El criterio de Goodman no protege contra esta rotura por fluencia. Para verificarlo, se calcula el momento máximo de Von Mises y se compara con el límite elástico.

$$\sigma'_{max} = [(\sigma_m + \sigma_a)^2 + 3(\tau_m + \tau_a)^2]^{1/2} < S_y$$

Para una comprobación rápida, basta con sumar σ'_a y σ'_m y comprobar que es menor que el límite elástico. En el peor caso, el punto D, $\sigma'_a = 46,41$ MPa y $\sigma'_m = 22,46$ MPa, entonces tomando el límite elástico 200 MPa:

$$46,41 + 22,46 < \sigma_y$$

Por lo tanto, no fallarán por fluencia.

3.1.11 Diseño de la unión del motor con el eje del sinfín.

Para poder llevar a cabo la transmisión de potencia hay que conseguir que el motor y los engranajes giren solidarios. Hay que idear un sistema mecánico que una motor y eje de una manera eficiente y duradera. Se va a sacar provecho de que la sección de los dos ejes (motor y clavija), tiene forma de D para introducirlos en un cilindro y apretarlos por la parte plana para que no puedan girar por separado.

El sistema está compuesto por 3 piezas, un cilindro de aluminio al que se le introduce por un extremo el eje del motor y dos tornillos o prisioneros M2 que aprietan los ejes una vez dentro del cilindro.

Lo más crítico es determinar el espesor mínimo del cilindro para que aguante la carga de torsión a la que se le va a someter, teniendo en cuenta que los prisioneros tienen que roscarse. El cilindro se comporta como un eje que está sometido a una carga de torsión $M_{motor} = 17,9$ Nmm, que aumenta y disminuye según cambia la afinación, pero se puede suponer que es constante para el punto de máximo esfuerzo y utilizar un coeficiente seguridad apropiado.

La tensión máxima aparece en el exterior del cilindro (cuando el radio sea máximo), y se calcula:

$$\tau_{max} = \frac{M_{torsión}}{I_o} \cdot R_{max}$$

Donde I_o es el momento de inercia, para una barra cilíndrica de radio R con un taladro interior de radio r:

$$I_o = \frac{\pi}{2} \cdot (R^4 - r^4)$$

Entonces la tensión máxima queda:

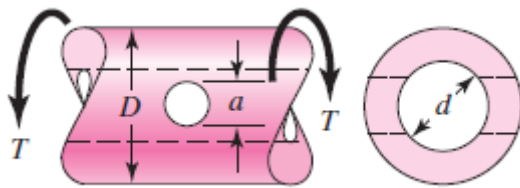
$$\tau_{max} = \frac{2M_{torsión}}{\pi(R^4 - r^4)} R$$

Los dos taladros roscados para los tornillos se comportan como posibles concentradores de tensiones, por lo que hay que incluirlos en los cálculos, hay que verificar que:

$$K_{fs} \cdot \tau_{max} = \frac{S_y}{n}$$

Donde n es el coeficiente de seguridad, S_y es el límite elástico y K_{fs} es el coeficiente de concentración de tensiones.

El valor de K_{fs} se calcula con las tablas siguientes:



a/D	0.9		0.8		0.6		0.4		0	
	A	K_{ts}	A	K_{ts}	A	K_{ts}	A	K_{ts}	A	K_{ts}
0.05	0.96	1.78							0.95	1.77
0.075	0.95	1.82							0.93	1.71
0.10	0.94	1.76	0.93	1.74	0.92	1.72	0.92	1.70	0.92	1.68
0.125	0.91	1.76	0.91	1.74	0.90	1.70	0.90	1.67	0.89	1.64
0.15	0.90	1.77	0.89	1.75	0.87	1.69	0.87	1.65	0.87	1.62
0.175	0.89	1.81	0.88	1.76	0.87	1.69	0.86	1.64	0.85	1.60
0.20	0.88	1.96	0.86	1.79	0.85	1.70	0.84	1.63	0.83	1.58
0.25	0.87	2.00	0.82	1.86	0.81	1.72	0.80	1.63	0.79	1.54
0.30	0.80	2.18	0.78	1.97	0.77	1.76	0.75	1.63	0.74	1.51
0.35	0.77	2.41	0.75	2.09	0.72	1.81	0.69	1.63	0.68	1.47
0.40	0.72	2.67	0.71	2.25	0.68	1.89	0.64	1.63	0.63	1.44

Figura 3.26 Factores aproximados de concentración de esfuerzo de un tubo con un taladro transversal sometido a torsión

El taladro interior tiene 4 mm de diámetro, lo determinan el diámetro del eje del sinfín 3,75 mm y el eje del motor 4 mm. El radio exterior es el que hay que calcular teóricamente, como las ecuaciones están bastante acopladas, se va calcular haciendo prueba y error, empezando a partir de 7 mm, que hace el espesor mínimo para que se puedan roscar los tornillos. Una regla básica a la hora de elegir la longitud de roscado para un tornillo es que esta sea mayor o igual que el valor de la métrica, si el tornillo es M6, hay que roscar al menos 6 mm:

$$\tau_{max} = \frac{2M_{torsión}}{\pi(R^4 - r^4)} R = 4,83 MPa$$

Donde $M_{torsión} = 17,9 Nmm$, $R = 3,5$ y $r = 2$.

K_{fs} se obtiene de la tabla:

$$\frac{a}{D} = \frac{2}{7} = 0,28 \rightarrow K_{ts} = 2,18$$

Para quedarse del lado de la seguridad, en vez de ajustar K_{fs} , se toma igual a K_{ts} , (el coeficiente de concentrador de tensiones teórico generalmente es mayor que el ajustado) entonces, si se toma como límite elástico del aluminio 60 MPa, el coeficiente de seguridad es:

$$n = \frac{S_y}{K_{fs} \cdot \tau_{max}} = 5,7$$

Por lo tanto el radio exterior es válido. La geometría de estas seis piezas es:

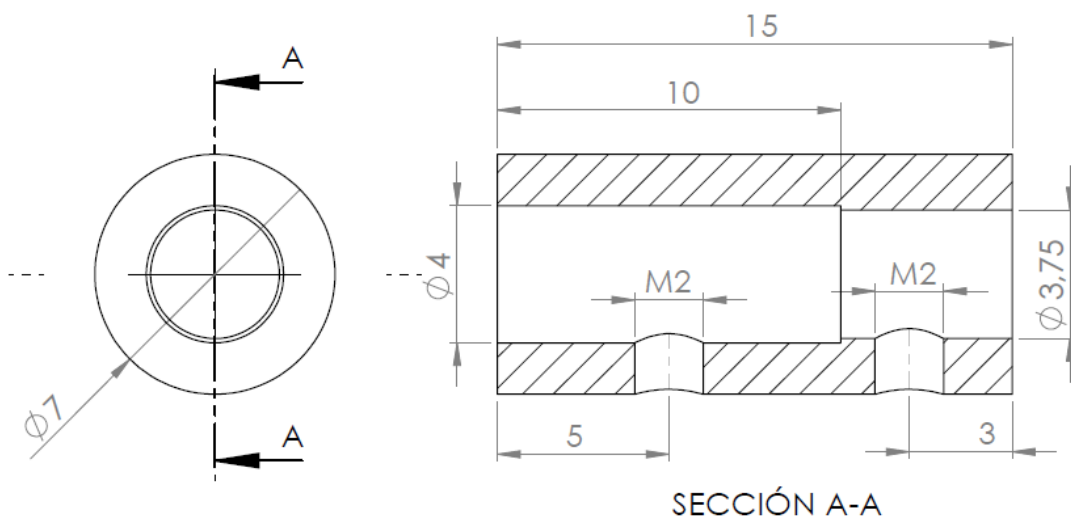


Figura 3.26 Plano de la pieza que une el eje del motor con la clavija de la guitarra.

3.2 Electrónica

3.2.1 Descripción del sistema y partes.

En este apartado se van a explicar los diferentes componentes electrónicos, su funcionamiento, y las diferentes conexiones necesarias para el funcionamiento del mismo.

El sistema electrónico consta de 4 partes.

1. Alimentación, formado por una batería o pilas y un regulador
2. Adquisición de datos y acondicionamiento de la señal, se utilizará un micrófono o un sensor piezoeléctrico y una etapa de amplificación.
3. Control del sistema: del control del afinador se encarga un microcontrolador.
4. Interfaz, formado por un grupo de LEDs y pulsadores.
5. Drivers y motores.

La primera decisión que se ha tomado es utilizar la plataforma arduino como microcontrolador, ya sea arduino UNO, arduino MEGA o arduino DUE. Debido a que arduino es muy sencillo de utilizar, y tanto el MEGA como el DUE son bastante potentes.

La figura 3.27 muestra un esquema de cableado general, con todas las conexiones necesarias para que el sistema funcione. Sirve como referencia tanto como para fabricar los cables como para colocarlos en su lugar, por lo que va a ser vital en la etapa de montaje del afinador.

Cada elemento está representado como una “caja negra” con todas sus entradas y sus salidas, más adelante se entrara en detalle de lo que contiene cada bloque, sus componentes y su funcionamiento.

Además también están explicados y dibujados los circuitos electrónicos con sus componentes. Algunas de las PCBs que se van a usar para el afinador se van a comprar directamente, las que haya que fabricar de forma manual se van a montar en placas de puntos, debido a que son placas simples de pocos componentes.

Para empezar, se van a describir cada una de las partes del circuito:

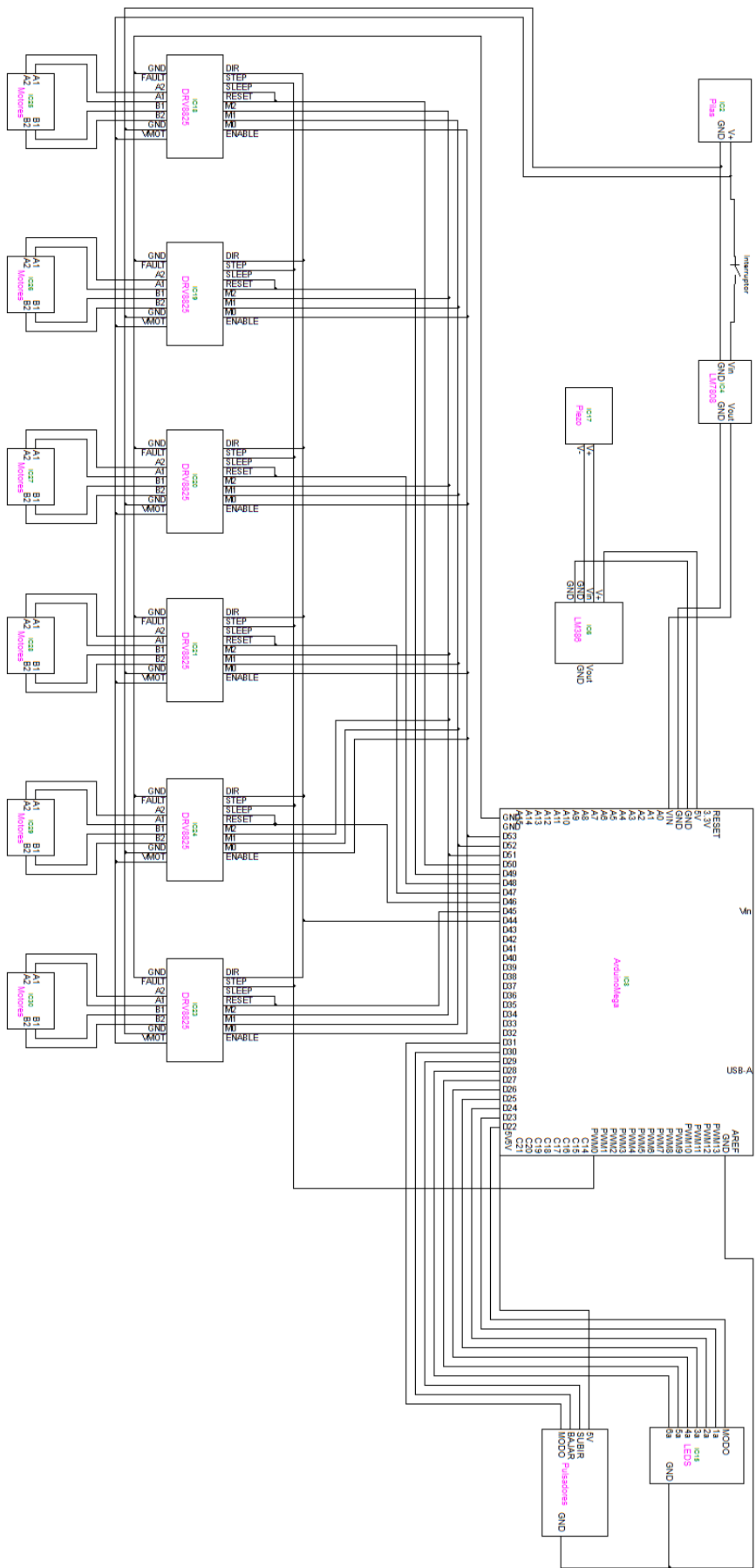


Figura 3.27 Esquema general de cableado del afinador automático

3.2.2 Alimentación y adquisición y acondicionamiento de la señal:

La siguiente imagen es una ampliación del esquema de cableado general, que corresponde a la alimentación del circuito (las dos primeras cajas de la izquierda) y a la toma y tratamiento del sonido:

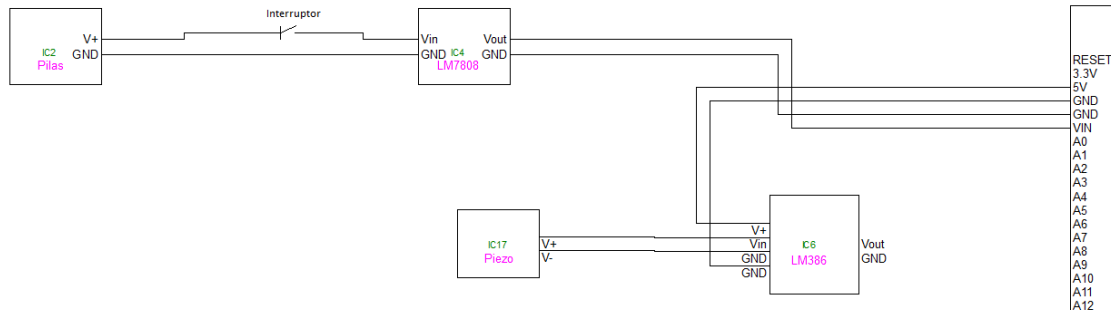


Figura 3.28 Detalle del esquema de cableado: alimentación, adquisición de la señal y amplificación del circuito

3.2.2.1 Alimentación

La alimentación del sistema la van a llevar cabo dos pilas de 9V, colocadas en serie, de forma que den 18V. La elección de este voltaje de funcionamiento se debe a dos razones:

1. Los motores funcionan entre 12 y 24 voltios. A 24V los motores consumen el doble que a 12, pero tienen mayor capacidad de carga, por lo tanto para alimentarlos se ha elegido un voltaje medio entre los anteriores de forma que aunque consuma, pueda dar el par necesario para un funcionamiento correcto
2. Se decidió no poner baterías porque para este rango de voltaje, las baterías que existen son de varias celdas. Esto, implicaría hacerse con un cargador específico para dicha batería y, además habría que diseñar un circuito que evitara la sobredescarga de la misma, ya que (por lo menos las baterías Li-Po) si se descargan por completo y después se cargan, se hinchan la primera vez y la segunda pueden explotar. Esto, hace que las baterías no sean la solución ideal para esta etapa del proyecto, debido a la mayor complicación y el coste (a corto plazo), así que por sencillez se van a usar pilas puestas en serie y para conseguir un voltaje entre 12 y 24, dos pilas de 9V.

El resto de componentes del sistema funcionan a 5V, y se va a alimentar a través del arduino. La entrada de alimentación del arduino funciona entre 7 y 12 voltios por lo que para bajar el voltaje es necesario un regulador de tensión.

Este regulador es el LM7808, que es un regulador lineal de 3 pines. Un regulador lineal es un regulador de tensión basado en un elemento activo que trabaja en su “zona lineal” y opera como resistencia variable en un divisor de tensión para mantener el voltaje constante a la salida. Su voltaje máximo de entrada es 35V, y puede soportar una corriente de salida de mayor que 1A, que es más de lo que necesita el sistema. Como se verá en la figura 3.29, que es un esquema del circuito, se colocan dos condensadores a la entrada ($C9=0,33 \mu\text{F}$) y a la salida ($C9=0,1 \mu\text{F}$) del regulador para eliminar cualquier fluctuación de voltaje que pueda ocurrir.

3.2.2.2 Adquisición de la señal

De la adquisición de la señal se encarga un sensor piezoeléctrico. Es un tipo de transductor cuyo funcionamiento se basa en el efecto piezoeléctrico, que es un fenómeno físico que presentan algunos cristales naturales o sintéticos que carecen de centro de simetría (eje polar). Ante una deformación mecánica, aparece una diferencia de tensión entre las caras del cristal. Esta diferencia de tensión es proporcional a la aceleración en sentido perpendicular a la superficie de dichas caras y al área de las mismas.

Los transductores piezoeléctricos pueden medir un amplio rango de frecuencias, desde 1 Hz hasta 15 KHz, aunque existen sensores que exceden los límites. Son pequeños, compactos y de fácil instalación.

La principal ventaja de este tipo de sensor, frente a un micrófono, (que fue una alternativa estudiada para la adquisición de la señal), es que no capta el ruido ambiente, únicamente produce la diferencia de tensión con la vibración de la superficie sobre la que este colocado. Por lo tanto, no es necesario poner filtros de frecuencia y hace que el tratamiento de la señal sea más sencillo.

Pero, tiene como desventaja que capta únicamente la vibración de la madera, y por lo tanto la señal que produce es muy débil. Recordemos además, que la vibración se produce inicialmente en la cuerda, en un punto alejado del piezoeléctrico, por lo tanto, la intensidad de la onda de sonido donde se recoge la señal habrá disminuido. Por todo lo anterior es necesario amplificar la señal del piezoeléctrico.

El circuito de la figura 3.29, representa la etapa de alimentación, adquisición, amplificación y tratamiento de la señal:

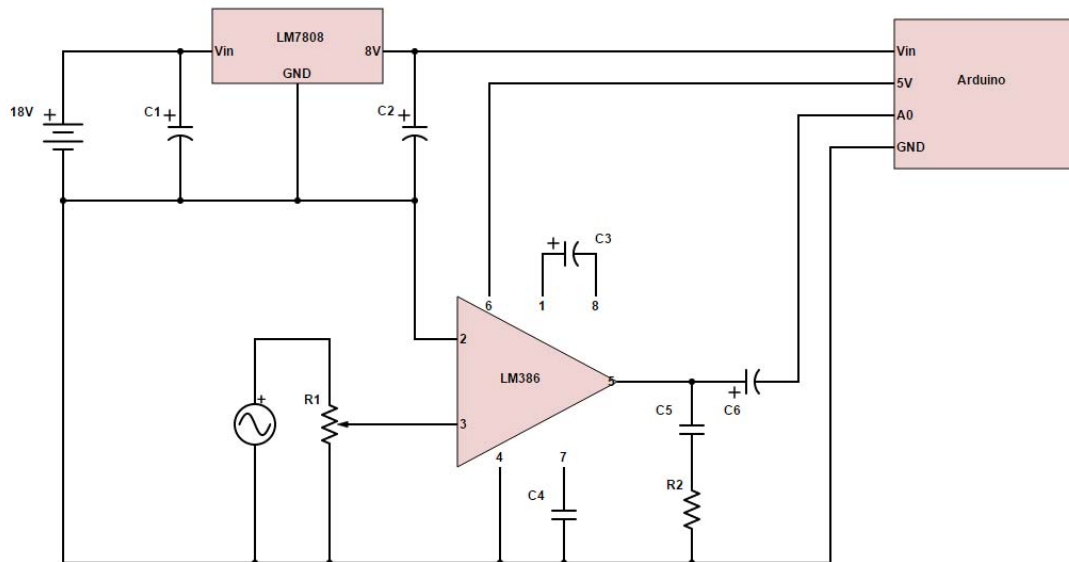


Figura 3.29 Circuito de alimentación, adquisición y amplificación de la señal

3.2.2.3 Amplificación y tratamiento de la señal

De la amplificación de la señal se encarga un amplificador de audio, el LM386. Tiene una serie de características que lo hacen ideal para el propósito para el que se utiliza:

1. Aunque trabaja con tensiones negativas, ya que la diferencia de tensión que viene del piezoeléctrico es una onda sinusoidal con valor medio 0V, se alimenta con tensiones positivas, en este caso 5V. Gracias a esto, puede haber una única "tierra" para todo el circuito y la conexión de las dos pilas es más sencilla, y no hay que sacar un cable entre las dos para referenciarlo a la tierra del amplificador.
2. La ganancia se puede cambiar colocando una resistencia y/o un condensador entre las patas 1 y 8, en este caso, con un condensador (C6) de 10 μ F, consigue una ganancia de 200.
3. Bajo consumo, alimentado a 6V consume 24 mW, lo cual es ideal para el funcionamiento con pilas. Tiene un potenciómetro a la entrada para control de volumen porque es un amplificador pensado para auriculares, pero ya que no se va a poder utilizar, se pone a volumen máximo.
4. Pese a que la alimentación es entre 5 y 0 voltios, la salida del amplificador está a 2,5V. Este offset es imprescindible para que el arduino pueda leer la señal amplificada, ya que la señal del piezoeléctrico es una señal "sinusoidal" que oscila en torno al cero y arduino solamente acepta valores de entrada positivos entre 0 y 5V. Además tensiones negativas podrían dañarlo.

3.2.3 Control del sistema.

Del control del sistema se encarga el Arduino Mega 2560. Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

Los microcontroladores son circuitos integrados programables que incluyen en su interior las unidades funcionales principales de un computador: CPU, memoria y periféricos de entrada/salida. El Arduino Mega, en concreto, tiene 54 entradas/salidas digitales de las cuales 15 pueden usarse para generar PWM y 16 entradas analógicas. Se programa mediante el software de Arduino que se puede descargar directamente de internet y utiliza un lenguaje de programación propio basado en el lenguaje de programación de alto nivel Processing que es similar a C++.

Se van a utilizar las siguientes entradas y salidas:

- Entrada analógica (A0): para convertir a digital la señal analógica amplificada proveniente del piezoeléctrico.
- Salidas digitales (D22-D28): estas siete salidas controlan los LEDs que indican la cuerda que se está afinando (6 leds) y el modo de afinador, automático o manual (1 led).
- Entradas digitales (D29-D31): Tres entradas para los pulsadores, que permiten interactuar con el programa.
- Salidas digitales (D44-D53): Estas diez salidas controlan los motores, el sentido de giro, el tamaño de paso, y que motor se tiene que mover cada vez.

- Salida digital (PWM1): esta es una salida digital especial, que crea una PWM 490 Hz de frecuencia y ancho de pulso configurable. Permite el giro de los motores paso a paso.

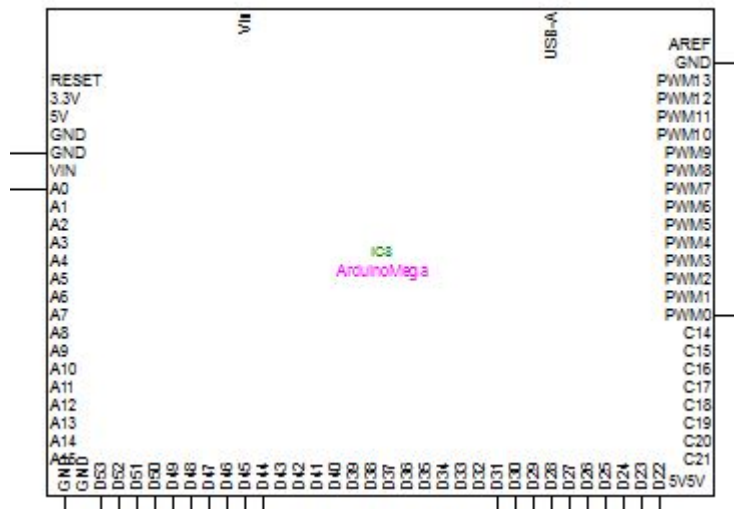


Figura 3.30 Detalle del esquema de cableado: Arduino MEGA

La figura 3.28 es una imagen del Arduino con todas las entradas y salidas que se van a utilizar. Aunque aún no se había indicado, todas ellas tienen que ir conectadas a un pin de tierra (GND).

3.2.4 Interfaz con el usuario.

3.2.4.1 Circuito LEDs

Para que el usuario pueda tener consciencia situacional de lo que está haciendo el programa y además interactuar con él, son necesarios algunos elementos hardware que sirvan de indicadores de estado y actuadores.

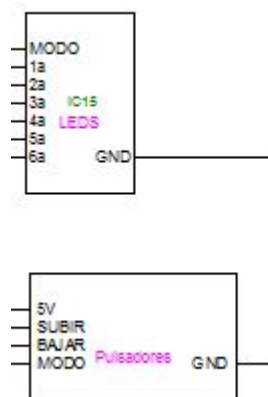


Figura 3.31 Detalle del esquema de cableado: LEDs y pulsadores

Los indicadores de estado son 7 LEDs, seis monocromáticos y un séptimo bicolor, los 6 de un color indican la cuerda sobre la que se está actuando, o se va a actuar, y el otro LED si el sistema está en modo manual o automático.

El circuito utilizado para los LEDs es bastante simple. Para calcular el valor de las resistencias de los LEDs, hay que tener en cuenta que están conectados a salidas digitales de 5 voltios cuando están en "HIGH". Sabiendo que la caída de tensión del led es 1,8V, y que la intensidad que va a circular por el LED se quiere fijar alrededor de 10 mA para no sobrecargar el arduino:

$$R = \frac{5V - V_{led}}{I_{led}} \approx 330 \Omega$$

Todas las resistencias tienen el mismo valor.

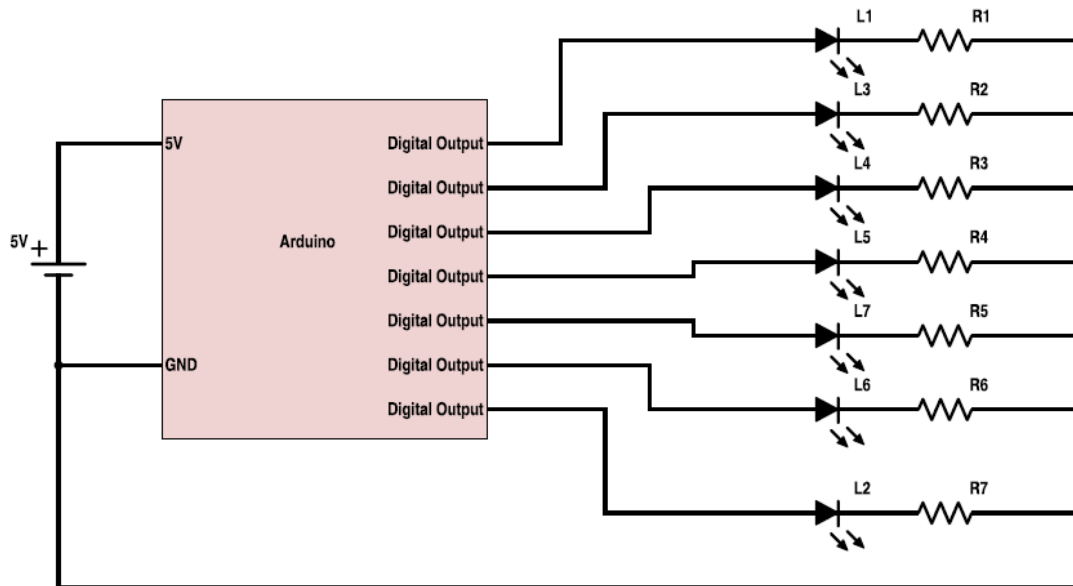


Figura 3.32 Circuito de control de los LEDs

3.2.4.2 Circuito pulsadores

Los tres pulsadores que se van a utilizar sirven para poder elegir la cuerda que se quiere afinar y el modo de afinación, el circuito también es sencillo:

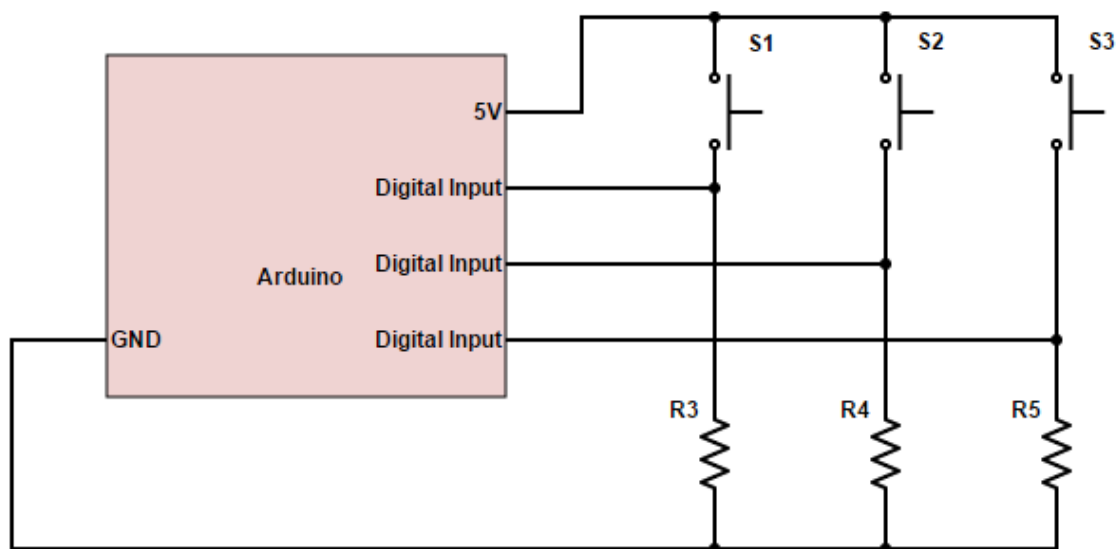


Figura 3.33 Circuito de control de los pulsadores

Para que los pulsadores funcionen hay que poner resistencias de pull-down, de 4,7K en este caso. Fuerzan "LOW" si el interruptor está abierto y "HIGH" si está cerrado. Si no se ponen estas resistencias, y se conecta directamente la entrada de arduino a un valor de tensión, cuando el interruptor está abierto se produce un estado de alta impedancia en la que el que la entrada está en estado indeterminado. Si se conecta a dos referencias de tensión (el circuito actual pero sin las resistencias) al cerrar el pulsador se produciría un cortocircuito que podría dañar el microprocesador.

Además, hay que tener en cuenta del efecto rebote de los pulsadores. Este efecto se produce cuando al cerrar un interruptor, aparece un rebote mecánico de sus contactos que no se puede evitar. Estos saltos producen más de un cierre del circuito, pudiendo convertir un único pulso en un número indeterminado de pulsos que serán considerados como datos a tratar. El efecto rebote puede evitarse colocando unos condensadores en paralelo con las resistencias de pull-down, o como se va a hacer en este caso, por software, poniendo un delay entre la lectura del estado del pulsador y la siguiente orden.

3.2.5 Drivers y motores.

3.2.5.1 Motores

La elección de los motores es un proceso complicado pero crucial para el buen funcionamiento del sistema, la primera decisión que hay que tomar es el tipo de motor eléctrico que se va a usar. Hay que elegir entre los motores eléctricos de continua, los servomotores o los motores paso a paso. Para empezar, los servomotores quedan descartados porque, aunque pueden mover cargas altas, tienen un recorrido muy pequeño, que no suele llegar a una vuelta, y aunque se pueda ampliar mecánicamente con engranajes el coste y el esfuerzo de diseño no merecen la pena habiendo alternativas mejores.

Los motores eléctricos de corriente continua, son motores baratos y fáciles de adquirir. Su funcionamiento se basa en la interacción entre el campo magnético de un imán permanente y el generado por la corriente que atraviesa las bobinas, ya sea una atracción o una repulsión hacen que el eje del motor comience su movimiento. Son muy útiles cuando la aplicación no necesita que el motor pare en un punto exacto, ya que estos motores tienen inercia al girar y no se detienen en seco cuando se desconectan de la fuente. En el caso del afinador, es vital que los motores dejen de girar en el mismo instante en el que deje de aplicárseles tensión.

Los motores paso a paso son, en definitiva, la mejor opción. Estos motores se caracterizan por moverse un paso cada vez que se les aplica un pulso a diferencia de los motores de continua que giran todo lo que dan cuando se les conecta a una fuente. Los pasos suelen variar dependiendo del motor desde 90° a 1,8°. Son más precisos y potentes que los de continua para el mismo tamaño, y además tienen par de frenado, que mantiene al motor fijo en su posición hasta que se le mande otro pulso que lo haga girar.

En el rotor tienen imanes permanentes y el estator bobinas excitadoras que se controlan externamente. Dependiendo del cableado de las bobinas existen dos tipos de motor paso a paso, los unipolares y los bipolares.

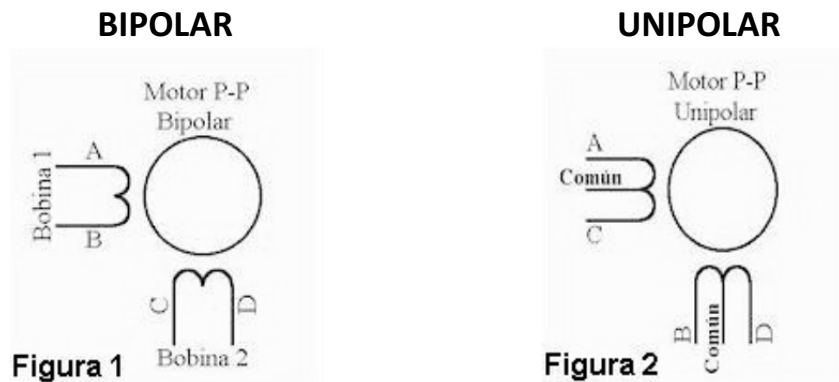


Figura 3.34 Esquema de cableado de un motor paso a paso bipolar y unipolar [19]

Los motores bipolares tienen por lo general 4 cables (A, B, C, D) son capaces de realizar mayores esfuerzos, por simple hecho de utilizar toda la bobina para mover el motor. Tienen la desventaja de que el control es más complicado, porque hay que invertir el flujo de corriente a través de las bobinas en la secuencia adecuada. Esto se consigue poniendo un puente H por bobina (un puente H es un circuito electrónico que debe su nombre a la representación gráfica del circuito, mediante 4 transistores permite invertir el sentido de la corriente).

Los motores unipolares tienen 5 o 6 cables, 4 están conectados a los extremos de las bobinas y el otro o los otros dos al centro de cada. Con esto se consigue que sean más fáciles de controlar, porque los dos cables centrales se conectan a V+ haciendo que la inversión de la corriente se vuelva sencilla. El inconveniente es que al utilizarse solo la mitad de la bobina se pierde capacidad de realizar par.

Se han elegido los motores bipolares por su mayor capacidad de carga.

Una vez elegido el tipo de motor, hay que determinar dos aspectos críticos, el par que puede desarrollar, y el tamaño del motor. El par necesario se ha calculado en el apartado de mecánica, dando como resultado 2 Ncm, y el tamaño máximo (ancho) del motor lo determina la separación de las clavijas de la guitarra, que es de 25 mm, por lo tanto, se puede ser flexible en cuanto al largo de los motores, pero no en el ancho y el alto (que suelen tener el mismo valor). Con estas restricciones, (ancho menor que 25 milímetros y par mayor que 2 Ncm), se han elegido unos motores de 4 Ncm de par y tamaño 20x20x40 mm aproximadamente. Son quizá demasiado largos pero cumplen las restricciones.

La desventaja de los motores paso a paso frente a los de continua, es que el control es bastante más complicado. El control no puede hacerse directamente desde el arduino porque no tiene suficiente potencia. Por lo tanto, se van a usar drivers para motor paso a paso.

3.2.5.2 Drivers

El driver elegido es el Dvr8825, es un driver para motores paso a paso bipolares, ampliamente utilizado para el control de impresoras 3D. Permite alimentar el motor desde 8 hasta 45 voltios, y puede llegar a entregar 2,5 amperios, con la debida refrigeración.

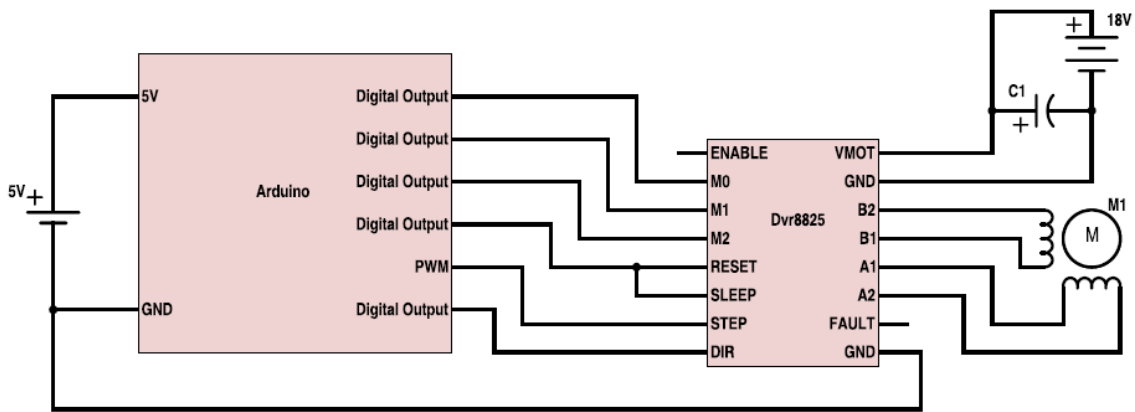


Figura 3.35 Circuito de conexión de un driver de motor paso a paso

Funcionamiento del driver, tiene 16 pines:

- ENABLE: un "HIGH" lógico inhabilita el indexador y las salidas del driver.
- MO, M1 y M2: permiten controlar mediante distintas configuraciones el tamaño de paso del motor. Así, se puede alcanzar una precisión de movimientos mucho mayor sin necesidad de variar la PWM que controla la velocidad de giro del motor.

MODE0	MODE1	MODE2	Microstep Resolution
Low	Low	Low	Full step
High	Low	Low	Half step
Low	High	Low	1/4 step
High	High	Low	1/8 step
Low	Low	High	1/16 step
High	Low	High	1/32 step
Low	High	High	1/32 step
High	High	High	1/32 step

Figura 3.36 Modos de configuración del driver

- RESET: cuando pasa de "high" a "low" inicializa el indexador lógico e inhabilita las salidas de los puentes H.
- SLEEP: "high" para habilitar el driver, si está en "low" entra en modo sleep, de bajo consumo.
- STEP: Un flanco de subida hace que el motor avance un paso.

- DIR: controla la dirección de giro.
- VMOT: alimentación del motor, a 18V.
- GND: tierra para la alimentación del motor.
- A1, A2, B1, B2: Conexiones de las bobinas del motor
- FAULT: es una salida de control, cuando hay un problema de sobrecalentamiento o sobrecorriente, la salida se pone en “low”.
- GND: tierra para las entradas de control.

Se van a conectar todas las entradas y salidas del driver menos ENABLE, que siempre está en LOW para que pueda funcionar, y FAULT, que no es necesaria porque el punto de trabajo del driver está alejado de los máximos de corriente y tensión, y no debería tener problemas de sobrecalentamiento ni sobrecorriente.

Como se puede ver en el esquema del cableado de los drivers, únicamente los pines RESET y SLEEP van a ser individuales para cada driver, estando conectados a salidas diferentes de arduino. Están cortocircuitados porque “trabajan” a la vez. El resto de pines de control (M0, M1, M2, DIR, STEP) van a ser comunes para todos los drivers y solo tendrán asignada una salida de arduino por pin para todos los drivers, con esto se consigue reducir el número de salidas de arduino diferentes a utilizar, y simplifica el programa de control al haber menos variables.

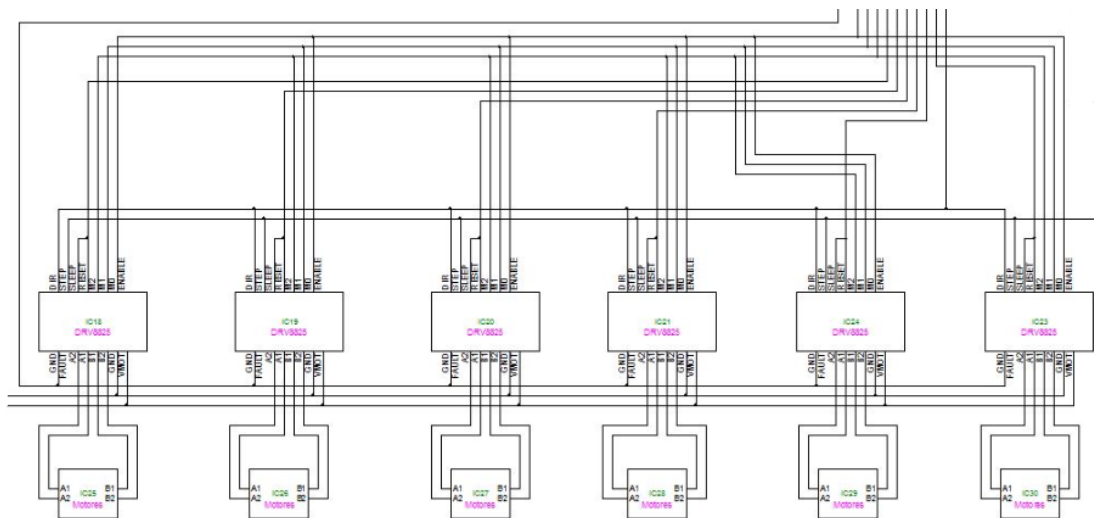


Figura 3.37 Detalle del esquema de cableado: Motores y drivers

En el circuito se ha puesto un condensador de entrada C1 de 100 μ F, que hace de reserva de energía en el caso de que el driver demande instantáneamente mucha corriente y para poder entregarla caiga el voltaje.

3.3. Software

3.3.1. Funcionamiento del programa

El objetivo del programa es que de manera automática o manual, la guitarra termine afinada. Hay unas limitaciones a tener en cuenta al hacer el programa, la primera es que por el elevado consumo de los motores, nunca debería funcionar más de uno a la vez. La segunda es que cada cuerda se debe afinar por separado. Todas las diferentes opciones que ofrece el programa se controlan por medio de tres pulsadores y se puede ver su estado por medio de 7 LEDs.

El funcionamiento del programa es sencillo, tiene tres modos, inicio, automático y manual.

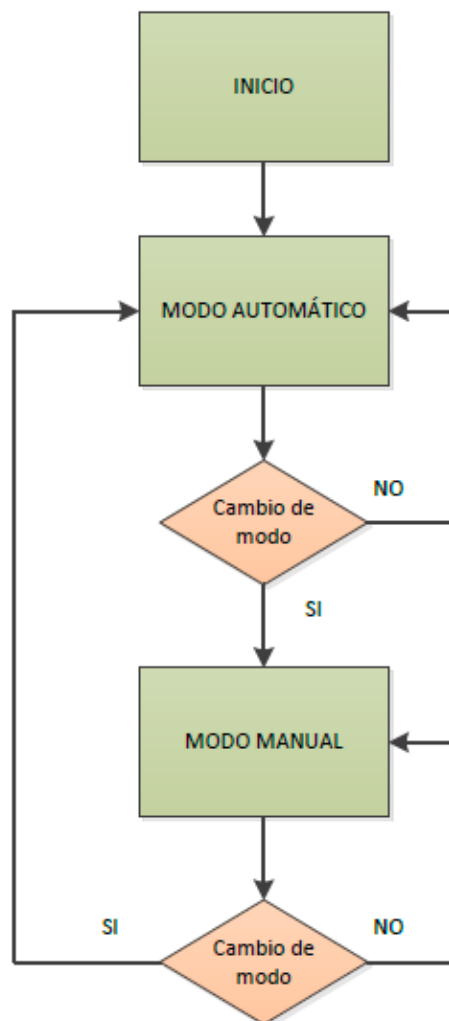


Figura 3.38 Diagrama de flujo general del programa

3.3.1.1 Modo Inicio.

Ocurre una vez en el transcurso del programa, cada vez que se encienda el arduino. Consiste en una secuencia de encendido y apagado de LEDs que dura entre dos y tres segundos y que no tiene ningún tipo de funcionalidad más allá de la estética.

3.3.1.2 Modo automático:

Una vez que termina el modo inicio, el programa pasa directamente a modo automático. Este modo consiste en: el programa mediante los LEDs indica la cuerda que se debe tocar, una vez que el usuario toque la cuerda, el programa reconoce la frecuencia que está sonando y eleva o disminuye la tensión de la cuerda (haciendo girar los motores) hasta conseguir la frecuencia deseada. Se puede saber que se está en modo automático porque el LED bicolor de modo esta fijo en rojo, y estará encendido el LED de la sexta cuerda, ya que por defecto, cada vez que se cambien de modo se va a empezar por esta cuerda.

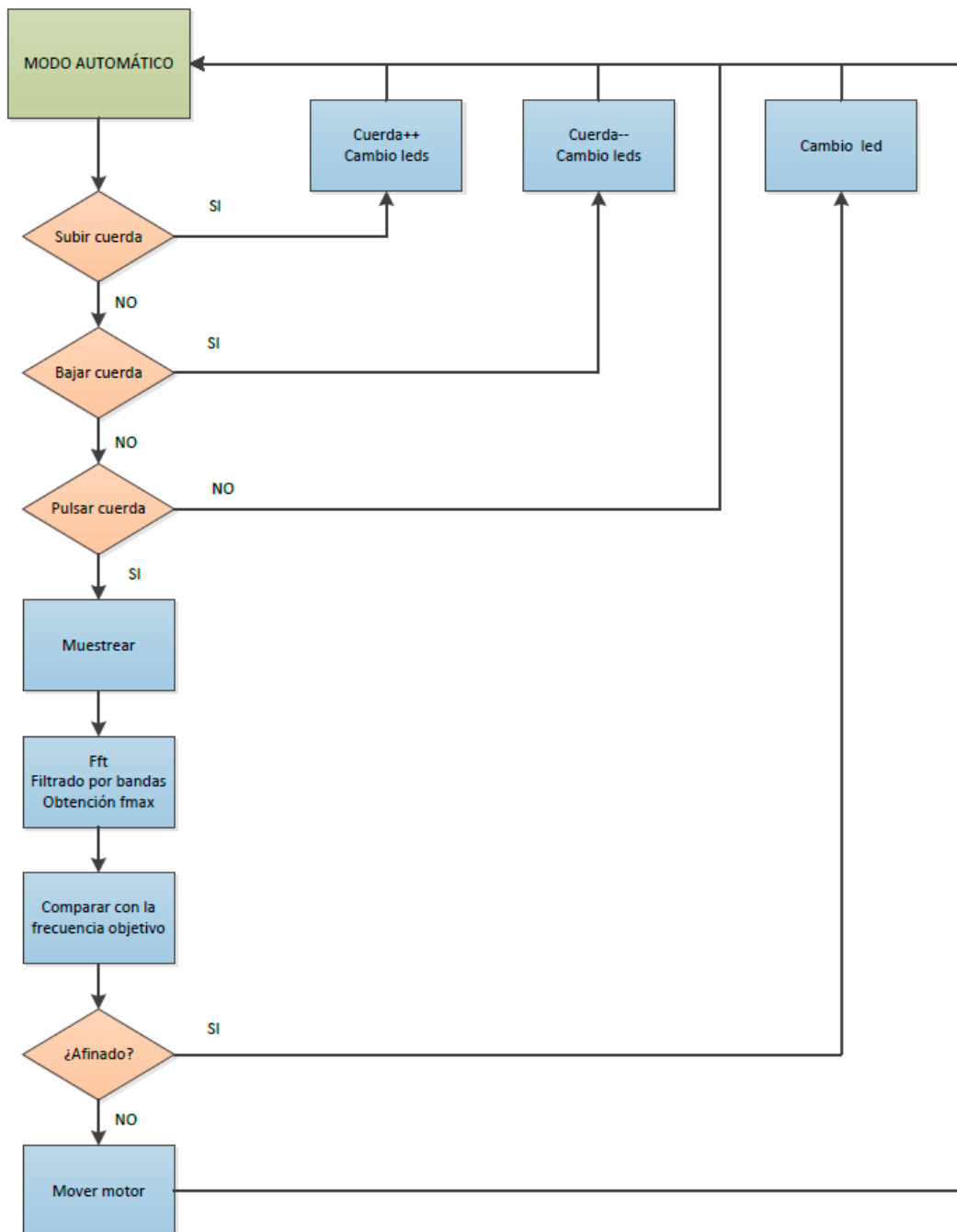


Figura 3.39 Diagrama del modo automático

Funcionamiento del modo automático: tras terminar el modo inicio, el programa se queda esperando a que el usuario realice una de las siguientes acciones:

1. Pulsar el botón de cambio de modo, en cuyo caso, sale del modo automático y pasa a manual. Para evitar el efecto rebote, nada más leer el valor del pulsador hay un delay de unos 200 milisegundos, que impiden que el arduino pueda leer valores erróneos. Este delay está presente en todas las lecturas de los pulsadores durante el programa.
2. Pulsar el botón de subir cuerda, o análogamente el de bajar cuerda, si se quiere afinar una cuerda diferente a la que se está afinando en ese momento, se puede cambiar de cuerda de una en una hasta llegar a la cuerda objetivo. Es decir, si se está afinando la quinta cuerda y se quiere afinar la primera, se puede pulsar el botón de subir cuerda tres veces, pasando por la cuarta y la tercera cuerda, o pulsar el de bajar y pasar por la sexta y la primera. (Si el programa está afinando la sexta cuerda y se pulsa el botón de bajar cuerda, pasa a la primera).
Siempre que se cambie de cuerda se enciende el LED de la cuerda a la que se cambia y se apaga el anterior, de esta forma se puede saber siempre que cuerda se hay que tocar y que motor que se va a mover.
3. Tocar la cuerda que se está afinando para que el programa pueda capturar la frecuencia y compararla con la frecuencia objetivo. Si no se toca la cuerda indicada, se inducirá un error en el programa.

Si se ha tocado una cuerda, el programa reconocerá la frecuencia y la compara con la frecuencia de la cuerda afinada, si la cuerda está afinada el LED de "modo" se pondrá verde durante un segundo, y si no está afinada el programa mandara a los motores girar un número de vueltas determinado.

3.3.1.3 Modo manual

Si estando en modo automático se pulsa el botón de cambio de modo, se pasa a modo manual. La existencia de este modo se debe a que, para acoplar los motores hay que quitar la parte de la clavija que se usa para girarla con la mano, de tal forma que una vez acoplados los motores se imposibilita cualquier giro manual. Así que, si no se pone este modo, se está perdiendo parte de la funcionalidad original de la guitarra. Se sabe que el programa está en modo automático si el LED de modo está de color verde. Por defecto, la primera cuerda sobre la que se actúa es la sexta, lo cual se indica encendiendo el LED de la sexta cuerda.

El funcionamiento del modo manual es sencillo:

1. Si se pulsa el botón de cambio de modo, se pasa a modo automático.
2. Si se pulsa el botón de subir cuerda de forma instantánea, el programa cambiara la cuerda sobre la que se actúa, de la sexta a la quinta, por ejemplo. Si se mantiene el botón pulsado durante más tiempo, el motor empezara a girar tensionando la cuerda hasta que el usuario suelte el botón.
3. Si se pulsa el botón de bajar cuerda ocurre lo mismo, pero el cambio de cuerda y el giro del motor son en sentido contrario.

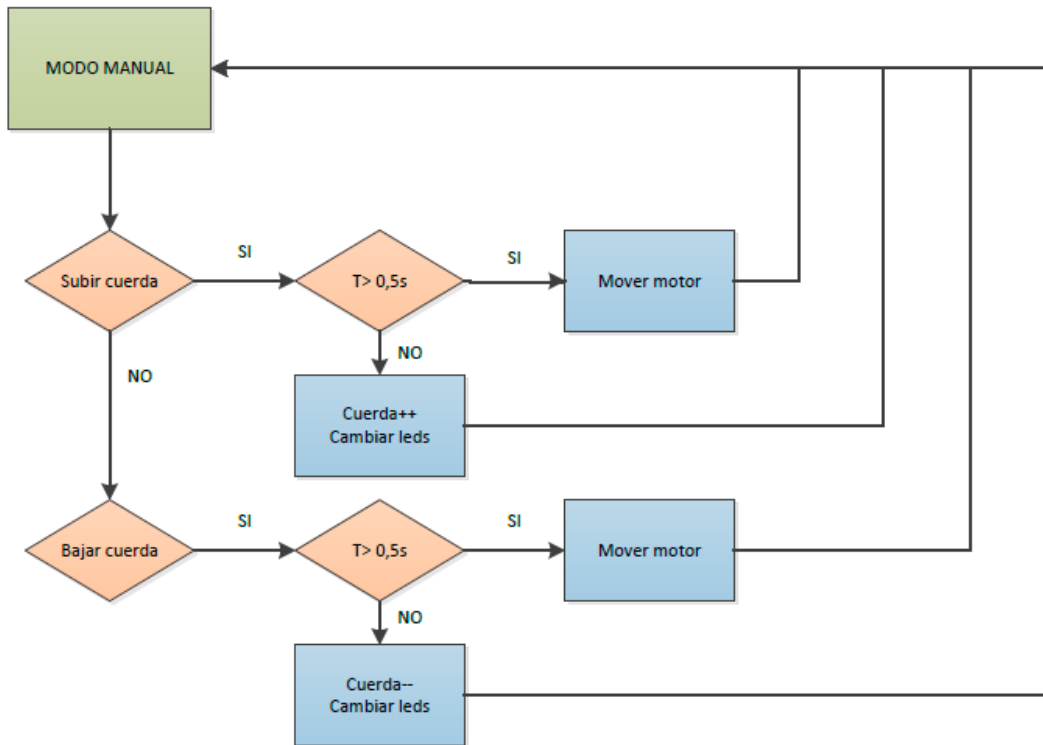


Figura 3.40 Diagrama del modo manual

3.3.2 Modo automático: Base teórica.

Se va a explicar cómo funcionan los bloques de código de: muestreo, transformada de Fourier obtención de la frecuencia fundamental y control de los motores.

3.3.2.1 Muestreo, frecuencia y número de muestras.

El sensor piezoeléctrico proporciona una señal variable en el tiempo de la que hay que estudiar su frecuencia. Hay que transformarla en una señal digital discreta para poderla analizar, de esto se encarga el conversor analógico digital (ADC). El ADC del arduino trabaja entre 0 cinco voltios con una resolución de 4mV, es decir, si lee un 0V escribe un valor entero 0, y si lee 5V escribe 1024.

Cuando no se está tocando ninguna cuerda, el ADC está leyendo continuamente valores próximos a 512 con una oscilación de +/-2 voltios. Se considera que se está tocando la cuerda indicada (por medio del LED), si la lectura del ADC es mayor que 512+100. Si esto ocurre, el ADC toma un número de muestras determinado que se almacenan en un vector para poder hacerles la transformada de Fourier. El valor "100" en "512+100" es orientativo, se necesita que sea lo suficientemente alto como para evitar que empiece a muestrear por algún pico transitorio de tensión debido a ruidos, pero no tan alto como para que el valor máximo de la señal al tocar una cuerda no sea capaz de superarlo, en cuyo caso, jamás se tomarían muestras.

El número de muestras a tomar lo determina el algoritmo FFT (transformada rápida de Fourier) que calcula la transformada discreta de Fourier (DFT), ya que tiene algunas limitaciones como que el número de muestras debe ser igual a una potencia de dos. La mayoría de los algoritmos

permiten la transformación de 512, 1024, 2048 o 4096 muestras. En este caso se usa una librería de arduino llamada “Arduino FFT Library”, que solo permite coger 256 muestras, lo que, como se verá ahora, conlleva una gran limitación.

La frecuencia de muestreo la determinan la frecuencia del sonido a adquirir y el número de muestras que se tomen.

Por un lado, según el teorema de muestreo de Nyquist-Shannon para poder adquirir toda la información de una señal periódica, es necesario muestrear al menos al doble de la frecuencia de dicha señal. Si no se satisface este criterio se produce un fenómeno llamado aliasing, por el que, debido a la baja frecuencia de muestreo la señal se puede confundir con una de frecuencia más baja, como puede apreciarse en la siguiente imagen.

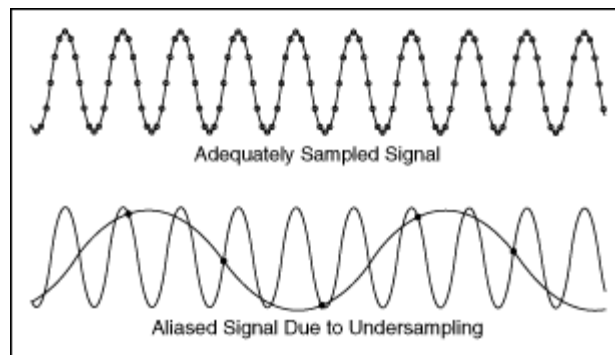


Figura 3.41 Fenómeno de aliasing [20]

Por otro lado, la resolución de la transformada de Fourier y el rango de frecuencias cubierto dependen directamente de la frecuencia de muestreo y del número de muestras.

$$\Delta f = \frac{f_s}{N}$$

Donde Δf es la resolución en Hz f_s es la frecuencia de muestreo y N el número de muestras. La resolución de la transformada es la separación en hertzios entre cada dato de la misma.

Se puede ver en el siguiente ejemplo, que si la frecuencia de muestreo es mucho mayor que el número de muestras, al captar bajas frecuencias se comete un error porcentual muy alto. Haciendo un análisis rápido, se puede ver que si la frecuencia de muestreo es por ejemplo 4096 muestras/segundo, y el número de muestras 128, la resolución sería de 32 Hz, (es decir, primer valor obtenido al realizar la transformada correspondería a 0 Hz, el segundo a 32, el tercero a 64, etc..) teniendo en cuenta que la frecuencia más alta de una guitarra (tocando las cuerdas al aire) es alrededor de 330 Hz, se cometería un error de hasta el 5% en cada medida, error que iría creciendo con el resto de cuerdas ya que vibran a menor frecuencia. Entonces, para cometer el menor error posible, hay que muestrear a la frecuencia mínima que cumpla el teorema de Nyquist.

Hay que fijar una frecuencia de muestreo, teniendo en cuenta lo anterior. La primera cuerda de la guitarra se afina a 329,6 Hz, si se tiene en cuenta que pueda estar sobretensada, podría llegar hasta los 350Hz. Por el teorema de Nyquist habría que muestrear al menos a 700

muestras/segundo. Si se deja un margen de seguridad, se va a fijar la frecuencia de muestreo alrededor de 1000 Hz, lo que hace una resolución de aproximadamente 4 Hz.

La “resolución” del oído humano, es de aproximadamente 1 Hz, por lo tanto, la resolución de 4 Hz al hacer la transformada y el error asociado de 2Hz por medida, no sería (estrictamente hablando) válido para afinar la guitarra. Afortunadamente, el resto de cuerdas tienen una frecuencia menor, y se puede ir bajando la frecuencia de muestreo para conseguir errores más bajos.

Se ha decidido usar tres frecuencias de muestreo diferentes: para la primera y la segunda cuerda (329,6 y 246,9 Hz respectivamente) se muestrea a 1000 Hz con un error de 2 Hz; la tercera y la cuarta cuerda (196 y 146,8 Hz respectivamente) a 600 Hz con un error de 1,2Hz aproximadamente; y la quinta y la sexta cuerda (110 y 82,4 Hz respectivamente) a 300 Hz con un error de 0,7 Hz.

3.3.2.2 Transformada de discreta de Fourier.

Es la herramienta principal del programa para detectar la frecuencia de la onda de entrada y realizar posteriormente una comparación con las frecuencias asociadas cada una de las notas.

La transformada de Fourier descompone la señal de entrada en una serie de senos y cosenos que permite obtener el espectro de frecuencias de una función, es decir, en vez de la amplitud de la señal en función del tiempo, al aplicar la transformada de Fourier se consigue tener el peso de cada frecuencia (que depende de la amplitud) en función de la propia frecuencia. Se puede apreciar en el ejemplo: estas graficas dibujadas con Matlab, se obtienen al caracterizar el comportamiento del piezoeléctrico en condiciones casi ideales, que se explicaran más adelante. La gráfica de arriba representa la señal en función del tiempo obtenía al tocar simultáneamente la primera y la segunda cuerda. La de abajo es la transformada de Fourier de dicha señal.

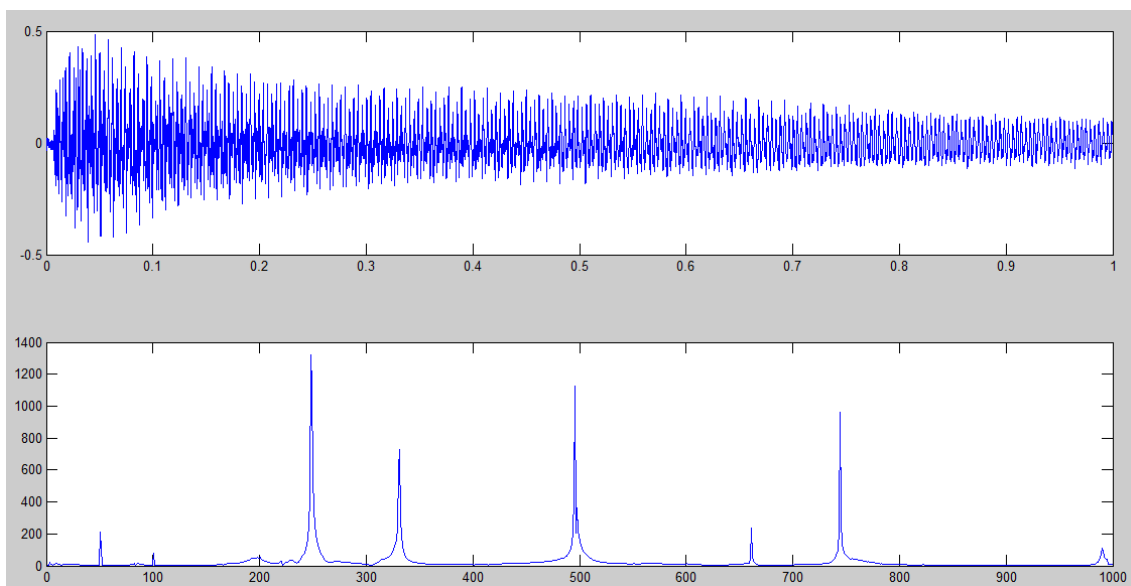


Figura 3.42 Graficas resultantes de grabar el sonido de las dos primeras cuerdas y su transformada de Fourier

Como se puede apreciar, aparecen distintos máximos de la función, los tres picos más altos, a 250, 500 y 750 Hz, corresponden a la segunda cuerda y a sus armónicos, el pico a 330 Hz y sus

armónicos a 660 Hz y 990 Hz a la primera cuerda, y los dos picos a 50 y 100 Hz corresponden a la frecuencia de la red y su armónico.

3.3.2.3 Algoritmo de obtención de la frecuencia fundamental.

La obtención de la frecuencia fundamental del sonido recogido es quizá la parte más importante del programa. Antes de empezar el proyecto existían una serie de dudas sobre como de “limpia” sería la señal obtenida con el piezoeléctrico, y si la muestra de mayor valor en la transformada de Fourier correspondería con la frecuencia fundamental. Para ello se hicieron una serie de experimentos en las condiciones más ideales posibles para determinar empíricamente el comportamiento del sensor piezoeléctrico, variando después dichas condiciones hasta llegar a las reales.

1. **Experimento 1:** Grabar una secuencia de una única nota en un archivo “.wav” para importarla y analizarla posteriormente con MATLAB. Para ello se usara el software CUBASE 5, una tarjeta de sonido externa (Focusrite Scarlett 2i2), un sensor piezoeléctrico y una guitarra clásica, que tiene mayor resonancia que una eléctrica y por tanto mayor vibración. La nota a analizar será el mi agudo (329,63 Hz), es decir la primera cuerda de la guitarra pulsada al aire, pero tapando las demás cuerdas para que no puedan vibrar inducidas por el sonido de la primera. El transductor se va a colocar en la pala de la guitarra y la tarjeta de sonido va a amplificar la señal del piezoeléctrico, porque es una señal muy débil (la tarjeta cuenta con un previo para micrófono).

Se ha creado ‘PruebaMatlab.wav’ que consta de 6 pulsaciones de la nota Mi. Se ha muestreado este archivo a 44100 hz, durante un segundo, obteniendo:

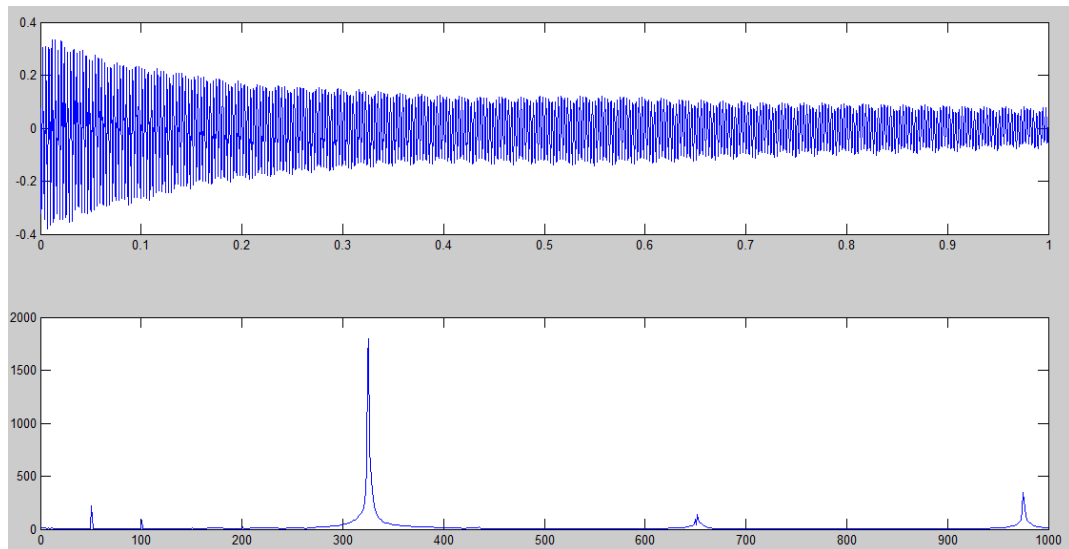


Figura 3.43 Grafica resultante de grabar el sonido de la primera cuerda y su transformada de Fourier

Como se puede ver, no existen más ruidos que el de la red eléctrica a 50 Hz, que además debería desaparecer al alimentar el sistema con baterías. El resultado obtenido es de gran importancia, porque corrobora la hipótesis de que con buscar el valor máximo de la transformada se obtiene la frecuencia fundamental. Además el espectro de frecuencias es el esperado, porque al contrario de lo que pasaría con un

micrófono, el piezoeléctrico no es capaz de detectar la vibración del aire, únicamente la de la madera, así que todos los ruidos externos del ambiente desaparecen y al haber tapado las demás cuerdas también desaparece cualquier tipo de ruido en la madera.

2. **Experimento 2:** Consiste en hacer exactamente lo mismo que antes, pero eliminando el software de grabación Cubase y tomando muestras en tiempo real. Con este experimento se pretende saber qué pasa si las muestras se toman entre una pulsación y otra de la cuerda, cuando la intensidad del sonido es mínima.

La programación se complica un poco porque para que Matlab reconozca la tarjeta de sonido, hay que declararla como un objeto y asignarle unas propiedades, etc...

Se muestrea a la misma frecuencia, y se toman nuevas muestras cada 0,5 segundos, en vez de graficar y teniendo en cuenta lo deducido en el experimento anterior, se ha añadido un algoritmo simple que busca la posición del máximo de la transformada de Fourier, y la muestra por pantalla.

El resultado del experimento es, como se podía esperar, que en cuanto decrece la intensidad del sonido de la guitarra, la frecuencia que se muestra por pantalla es la de la red. Cuando el sistema se alimenta con baterías/pilas debería mostrar la frecuencia 0. Si el programa tomara este valor como válido y empezara a mover los motores, la guitarra se desafinaría en vez de afinarse. Por lo tanto, hay que filtrar esos resultados, se puede hacer digitalmente programando que no se tengan en cuenta.

El programa utilizado para este experimento no solo es importante por su resultado. Este programa, si se le añaden unas pocas líneas de código, es un AFINADOR DIGITAL. El código está incluido en el Anexo B.2, por si en un futuro algún compañero quiere utilizarlo. Se le ha añadido un algoritmo que muestra por pantalla la frecuencia fundamental de la señal muestreada, y la frecuencia y el nombre de la nota más cercana a ella. De esta forma, se podría afinar cualquier instrumento con bastante precisión.

3. **Experimento 3:** Se repite el experimento 1, pero esta vez se va a cambiar la tarjeta de sonido por un arduino Mega y el amplificador de audio LM386 con ganancia de 200 y en vez de una guitarra española se usa la guitarra eléctrica sobre la que se va a montar el afinador. La función del arduino es únicamente recoger las muestras, que luego se pasan a un script de Matlab para realizar la transformada de Fourier. Este experimento tiene como objetivo caracterizar el comportamiento del amplificador, y ver que ruidos genera y amplifica. De la alimentación del amplificador se encarga una fuente de continua regulada y de la del arduino el ordenador (que al estar conectado para poder mostrar los valores por pantalla también lo alimenta).

La frecuencia de muestreo es 1024 muestras/segundo, que es la que se va a usar en el programa definitivo, pero para aumentar la resolución se han tomado 1024 muestras en vez de 256. La siguiente figura muestra la gráfica de la transformada de Fourier, como se puede observar, el pico de mayor amplitud corresponde a la frecuencia 0. Si se mira la forma de las series de Fourier, aparece un término debido a una constante a_0 :

$$\frac{a_0}{2} + \sum_{n=1}^{\infty} \left[a_n \cdot \cos \frac{2n\pi}{T} \cdot t + b_n \cdot \sen \frac{2n\pi}{T} \cdot t \right]$$

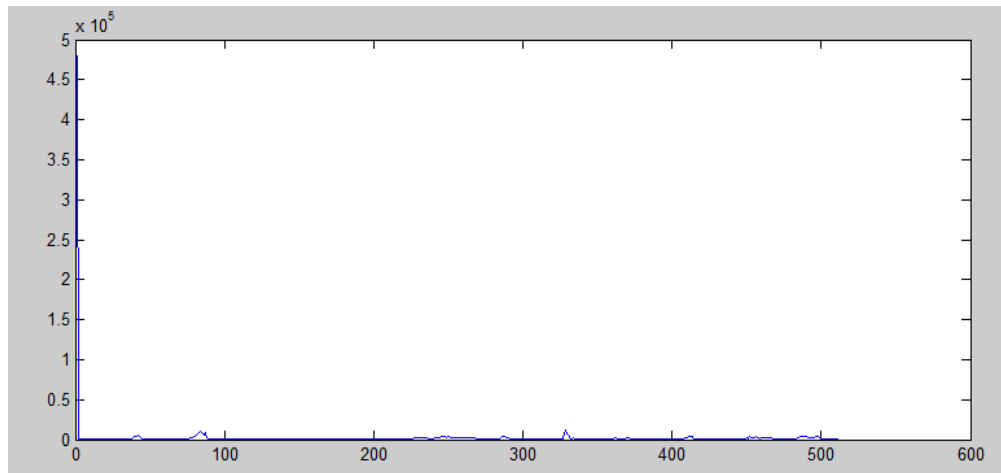


Figura 3.44 Transformada de Fourier del sonido de la primera cuerda capturado con Arduino

Si la señal fuera sinusoidal pura, este término no debería aparecer, pero, hay que añadir un offset de continua para que el arduino pueda leer la señal, ya que solamente admite tensiones positivas. Esto es lo que causa el máximo en el 0. Este máximo siempre va a estar presente y hace que el máximo de la transformada de Fourier no coincida con la frecuencia fundamental. Por ello y dado que la frecuencia más grave se encuentra alrededor de los 80 Hz, se van a quitar del algoritmo de búsqueda de la frecuencia fundamental (en el programa definitivo) las frecuencias que estén por debajo de los 55 Hz. La nueva grafica queda:

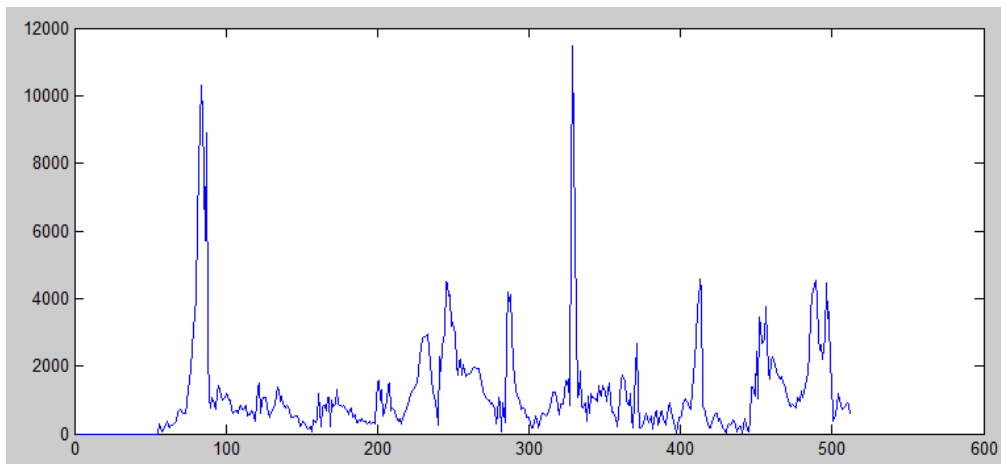


Figura 3.45 Transformada de Fourier del sonido de la primera cuerda filtrando bajas frecuencias

Nótese que en realidad es la misma grafica de antes, pero al haber eliminado el pico de frecuencia 0, la gráfica se ha reescalado. Ahora ya sí que salen resultados coherentes, el máximo se encuentra alrededor de los 330 Hz que corresponde con la cuerda tocada, y, como se puede ver hay bastante ruido por debajo y otro máximo en 90 Hz aproximadamente. El ruido se puede deber a no haber tapado las cuerdas de la guitarra al tocar, y el origen del segundo máximo podría deberse a la red y a un problema de precisión del arduino que por errores de redondeo transforme, al multiplicar, los 100 Hz en 90-95 Hz.

Para eliminar estos ruidos que podrían afectar a la medida de la frecuencia, se van a fijar unos rangos de frecuencias lógicos entre los que se movería cada cuerda en el caso de estar desafinada y son los que se van a usar para calcular la frecuencia fundamental. Por ejemplo: cuando una cuerda está por encima de su afinación (sin haber tocado las clavijas) suele ser debido a un cambio de temperatura, por lo que la variación será pequeña. Para la primera cuerda, se podría fijar el valor máximo del rango en 350 Hz. Sin embargo, por debajo influyen más factores aparte de la temperatura, como fenómenos de fluencia en la cuerda (con el paso del tiempo al estar tensionadas se van alargando) o de plastificación en algún punto debido a una sobreelongación de la cuerda al tocar (como un bending) que produzca una tensión por encima de la normal. En este caso si hay más variación de frecuencias y se podría considerar como límite inferior 240 Hz. En la siguiente grafica se puede ver el rango de frecuencias aplicado para la primera cuerda:

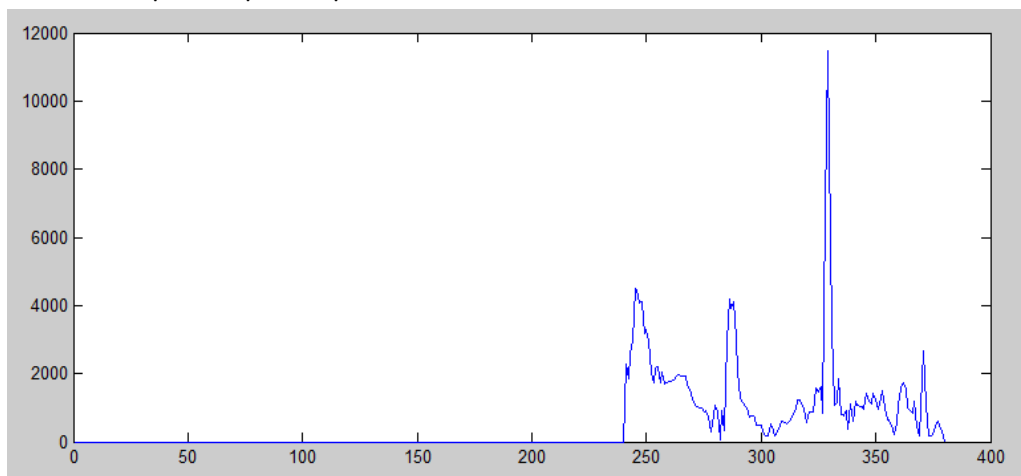


Figura 3.46 Transformada de Fourier del sonido de la primera cuerda filtrando por bandas de frecuencia

Esta forma de operar tiene como ventaja que los armónicos de la frecuencia fundamental que tienen bastante peso en la transformada (son máximos) y podrían llevar a error se eliminan, pero tiene como inconveniente que si no se toca la cuerda que indica el programa, se hacen medidas erróneas porque la frecuencia a medir no se encuentra dentro del rango de frecuencias entre las que se está buscando.

4. **Experimento 4. (programa definitivo).** Se elimina Matlab, y el Arduino se encarga de muestrear y aplicar la FFT. Juntando todo lo aprendido hasta ahora, se llega a qué:
 - La frecuencia de muestreo para las cuerdas tiene que variar dependiendo de la cuerda desde 1000 Hz hasta 300 Hz.
 - El número de muestras a tomar es fijo e igual a 256 y viene impuesto por el algoritmo de la transformada discreta de Fourier.
 - El algoritmo para encontrar la frecuencia fundamental únicamente busca el máximo de los valores de la transformada.
 - Para eliminar fuentes de error, el algoritmo de búsqueda de la frecuencia fundamental solo tiene en cuenta el rango de valores entre los que se suele encontrar la frecuencia de la cuerda.

Este experimento muestra por pantalla los valores de frecuencia obtenidos al muestrear, hacer la transformada, hacer el filtrado por frecuencias y buscar el máximo de la transformada.

En realidad, lo que interesa no es el valor del máximo, si no la posición de dato que contiene el máximo de los 256 que se obtienen al hacer la transformada. Si este número se multiplica por la resolución de la transformada: F_s/N , se obtiene el valor de la frecuencia fundamental del sonido. Este programa da buenos resultados en general, mejor para las cuerdas agudas que para las graves, probablemente porque el ruido de la red se solape con el sonido de la cuerda.

3.3.2.4 Control de los motores.

Una vez detectada la frecuencia fundamental, se compara con la frecuencia objetivo y se hace girar los motores hasta que las dos frecuencias se igualen. En un principio, el control del giro del motor se pensaba hacer mediante un controlador PID. Esta opción se descartó debido a que, sin complicar el programa, no se puede muestrear a la vez que se hace girar el motor y el control se vuelve muy lento y poco preciso.

Otra manera de hacerlo es caracterizando para cada cuerda cuanto varia la frecuencia a cada paso del motor. Esta opción es mejor en cuanto a precisión y rapidez, ya que con solo una medición casi se podría tener afinada la cuerda, pero conlleva un trabajo previo mucho mayor. Es la que se va a utilizar. Hay que generar unas tablas de datos que muestran la variación de frecuencia en función del tiempo que han girado los motores. Se parte de la tensión máxima que es capaz de aplicar el motor a la cuerda y se hace girar el motor a intervalos de tiempo iguales midiendo la frecuencia de la cuerda después de cada giro. Se va a suponer que la variación de frecuencia entre dos datos es lineal, de forma que se puede hacer una interpolación lineal para calcular el tiempo que debería girar el motor para llegar a la frecuencia objetivo.

4. Diseño en 3D, fabricación, montaje y pruebas

Una vez elegidos todos los componentes hay que estudiar la mejor manera de distribuirlos por la pala de la guitarra. Además hay que colocarlos de forma que los cables se crucen lo menos posible.

4.1 Diseño en 3D.

Para poder construir adecuadamente el afinador, se ha hecho un diseño en 3D del mismo. En él se han colocado todos los componentes comerciales y las piezas y PCBs de fabricación propia.

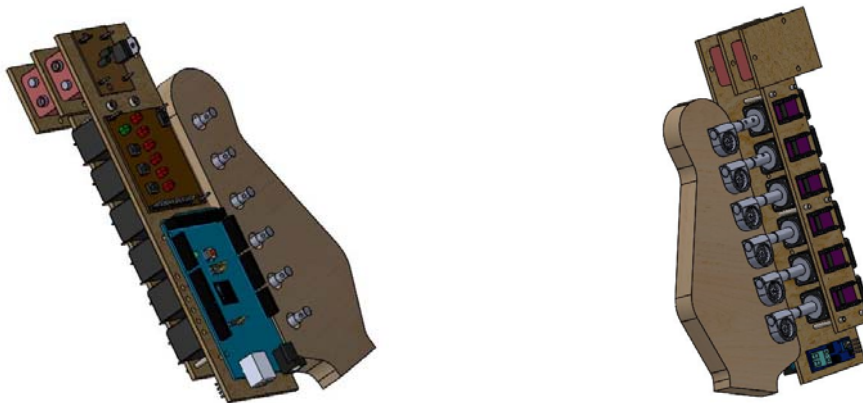


Figura 4.1 Diseño en 3D del afinador.

Las piezas de madera tienen dos funciones, por un lado, evitan que los motores giren sobre sí mismos cuando se intenta aflojar o apretar una cuerda, y por otro lado, sirven para sujetar los componentes y sacarlos fuera de la pala de la guitarra para evitar tener que mecanizarla o aplicarle pegamentos que estropearían su acabado.

La mayoría de las piezas van atornilladas y algunas PCBs, la del amplificador y los drivers, que no tienen espacio para hacerles taladros se van a pegar a la madera.

Existen unas dimensiones críticas para el montaje del sistema. La primera es la separación entre clavijas en la propia pala de la guitarra, limitan el ancho de los motores a menos de 25 mm, que es la separación entre clavijas. Para que el afinador no estorbe, no puede sobrepasar el saliente inferior de la pala, y por estética, tampoco debería sobresalir por la parte superior, esto limita el largo de las piezas de madera a 23 centímetros como máximo.

4.2 Fabricación

4.2.1 Fabricación de piezas mecánicas

Según el diseño, hay que fabricar 3 piezas para poder anclar todos los componentes del sistema. Son piezas planas con taladros para poder fijar los componentes con tornillos. En un principio se estudió la posibilidad de hacerlas de fibra de vidrio, porque para el mismo espesor es más resistente que la madera, pero, por la misma razón es más difícil de mecanizar. Además, hay que pegar algunos componentes a la superficie de las planchas, y la fibra de vidrio suele ser lisa. Por estas razones, facilidad de mecanizado y superficie porosa, se ha

elegido madera de contrachapado de 4 mm para hacer las piezas. Tienen las siguientes dimensiones:

- Pieza 1: Sirve para anclar el Arduino, la PCB con los pulsadores y los LEDs, los motores, las pilas, el regulador y el amplificador.

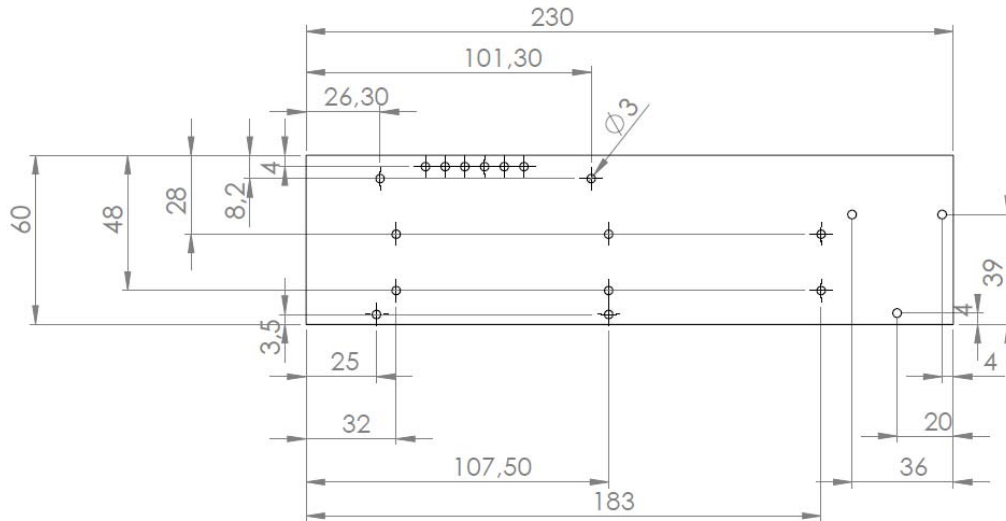


Figura 4.2 Plano de la Pieza 1

- Pieza 2: Sirve para anclar los motores, se utilizan 6 tornillos que van desde la pieza 1 a la 2 para evitar que los motores giren sobre sí mismos.

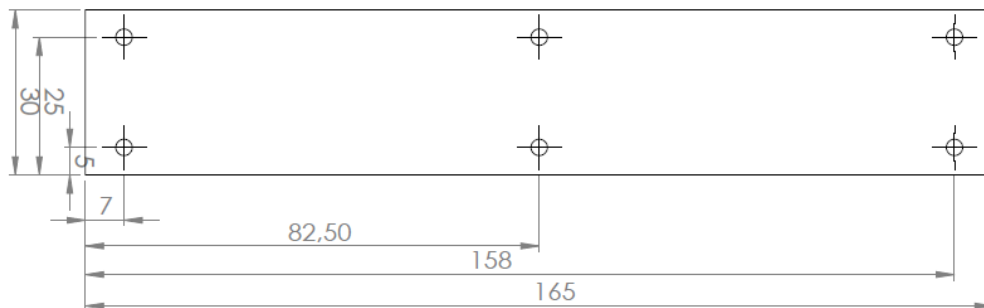


Figura 4.3 Plano de la Pieza 2

- Pieza 3: Sirve para sujetar las pilas.

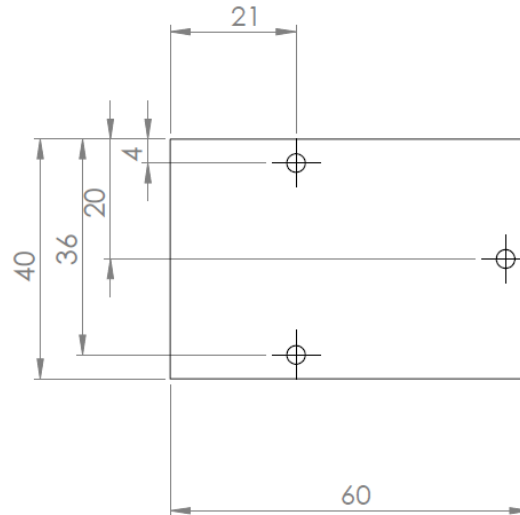


Figura 4.4 Plano de la pieza 3

Las piezas se han fabricado con un serrucho y un taladro, el resultado es:

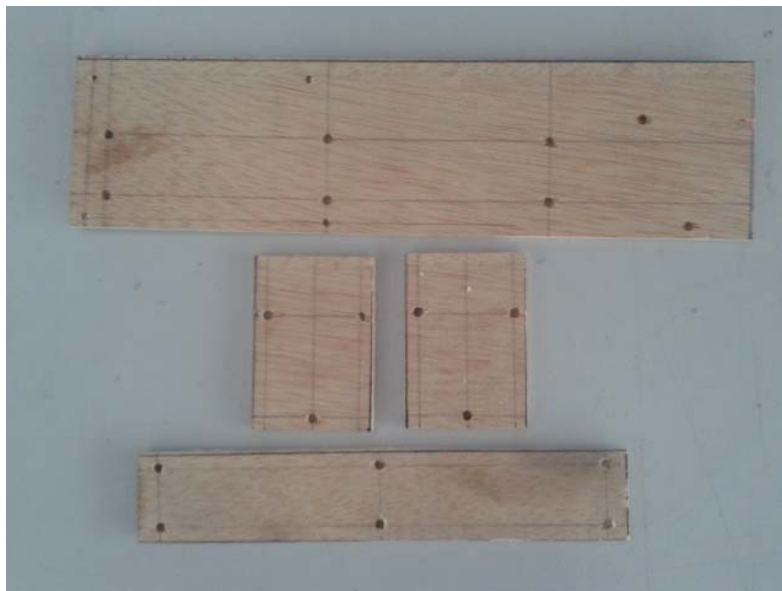


Figura 4.5 Piezas de madera

Las piezas que unen el eje del motor con el de la clavija se mecanizan a partir de una barra de aluminio de 7mm de diámetro, por medio de un torno manual y un taladro de columna. Son necesarias 5 operaciones, primero se hacen en el torno el taladro pasante de 16 mm con la broca de 3,75, luego se taladran 10 mm con la broca de 4. Se hace un tronzo a 15 mm del extremo y se lleva la pieza al taladro de columna para hacer los agujeros para los prisioneros a 3 y 10 mm. Por último se roscan con un macho de roscar M2.

- Unión ejes:

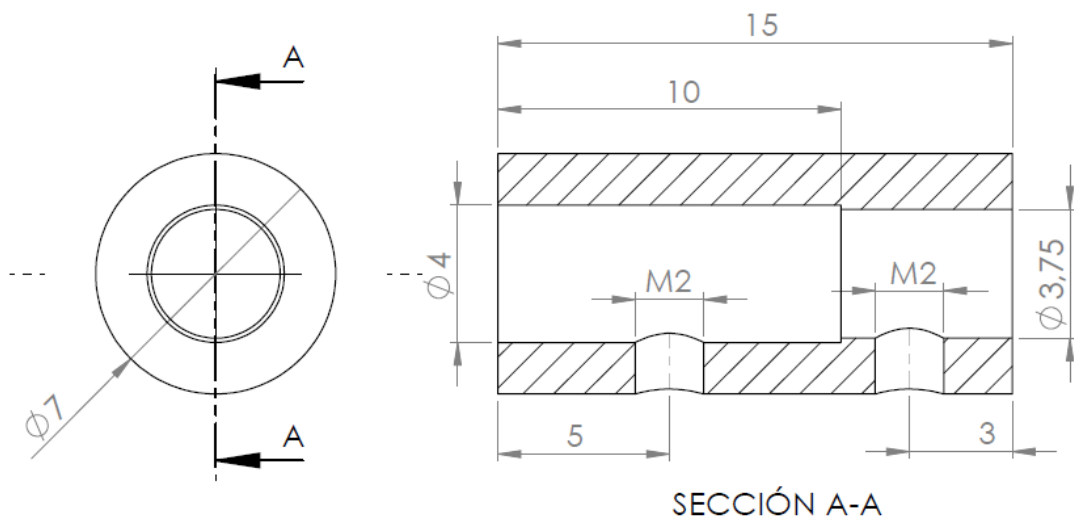


Figura 4.6 Plano de la pieza Unión ejes.

El resultado de la fabricación es:



Figura 4.7 Unión ejes

4.2.2 Fabricación de PCBs

En el circuito va a haber 4 tipos de PCB:

1. Amplificador Lm386: En un principio se intentó comprar los componentes por separado y soldarlos en una placa de puntos, después de quemar dos integrados se llegó a la siguiente conclusión, que se ha mantenido para el resto de PCBs: por relación calidad precio, es más barato comprar la PCB entera y soldada con todos sus componentes, que comprar los componentes por separado y soldarlos en una placa de puntos, además se gana en calidad estética y funcional.
2. Regulador LM7808: Se intentó comprar el regulador ya soldado, pero tenía un tiempo de entrega excesivamente alto y hubo que comprar los componentes por separado y soldarlos en una placa de puntos.

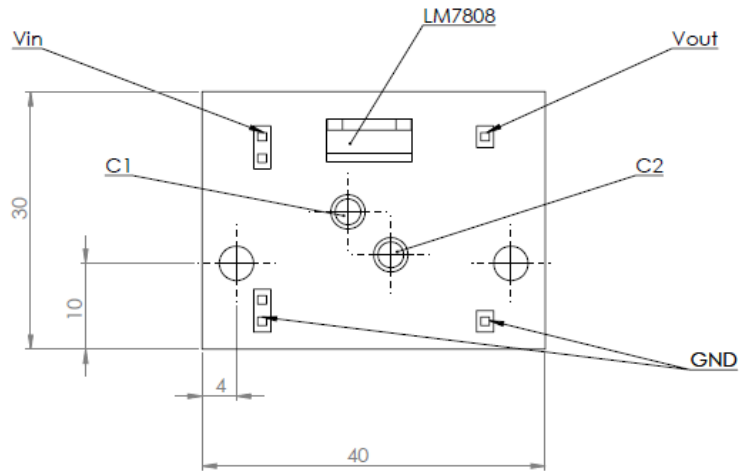


Figura 4.8 Placement de la PCB del regulador.

3. Driver: Se compra montado y soldado.
4. Pulsadores y leds: esta PCB hay que fabricarla porque no se puede encontrar en el mercado una placa con las mismas características. Todos los componentes tienen que caber en un espacio de 60x50mm debido a limitaciones físicas en el montaje. Para facilitar el montaje, antes de soldar los componentes se ha hecho un 'placement' de la PCB.

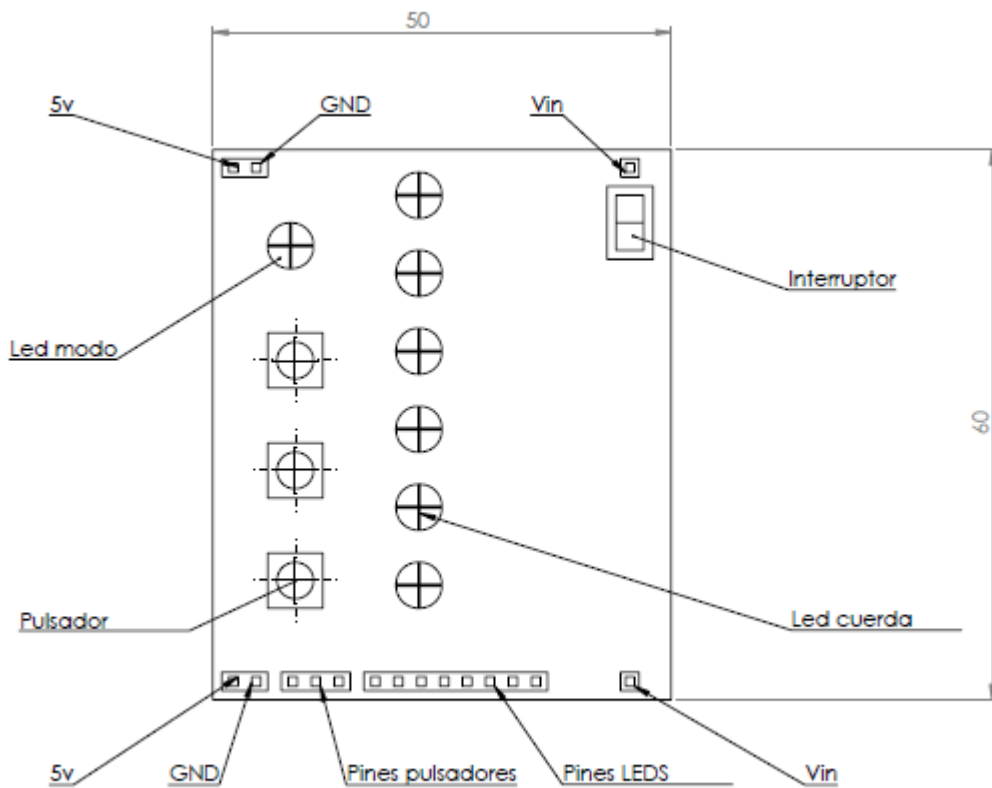


Figura 4.9 Placement de componentes de la PCB Interfaz con el usuario

4.2.3 Fabricación de cableado

La corriente que pasa por los cables varía entre los 10mA del circuito de los LEDs y más de 1A en los cables que van a los motores. Se han elegido cables de galga 24 que, como se puede ver en la figura, soportan en distancias cortas más de 3.5A.

Codigo AWG	Diametro del conductor (mm)	Ohmios por kilometro	Amperaje maximo para distancias cortas	Amperaje maximo para distancias largas
OOOO	11.684	0.16072	380	302
OOO	10.40384	0.202704	328	239
OO	9.26592	0.255512	283	190
0	8.25246	0.322424	245	150
1	7.34822	0.406392	211	119
2	6.54304	0.512664	181	94
3	5.82676	0.64616	158	75
4	5.18922	0.81508	135	60
5	4.62026	1.027624	118	47
6	4.1148	1.295928	101	37
7	3.66522	1.634096	89	30
8	3.2639	2.060496	73	24
9	2.90576	2.598088	64	19
10	2.58826	3.276392	55	15
11	2.30378	4.1328	47	12
12	2.05232	5.20864	41	9.3
13	1.8288	6.56984	35	7.4
14	1.62814	8.282	32	5.9
15	1.45034	10.44352	28	4.7
16	1.29032	13.17248	22	3.7
17	1.15062	16.60992	19	2.9
18	1.02362	20.9428	16	2.3
19	0.91186	26.40728	14	1.8
20	0.8128	33.292	11	1.5
21	0.7239	41.984	9	1.2
22	0.64516	52.9392	7	0.92
23	0.57404	66.7808	4.7	0.729
24	0.51054	84.1976	3.5	0.577
25	0.45466	106.1736	2.7	0.457
26	0.40386	133.8568	2.2	0.361
27	0.36068	168.8216	1.7	0.288
28	0.32004	212.872	1.4	0.226
29	0.28702	268.4024	1.2	0.182
30	0.254	338.496	0.86	0.142
31	0.22606	426.728	0.7	0.113
32	0.2032	538.248	0.53	0.091

Figura 4.10 Galga de los cables y corriente que soportan [21]

No se van a usar conectores, se va a poner pines macho en un extremo del cable para conectarlo a arduino, y hembra para el resto de conexiones a las PCBs. Esto permite liberar el arduino en caso de que quiera usarse para otra aplicación.

Los pines hembra se van a soldar a las PCBs para minimizar el riesgo de malos contactos y ruidos



Figura 4.11 Pines macho y hembra para conectar los cables

Siempre que se pueda, los cables se van a agrupar por mazos, bien sea con termorretractil, cinta aislante, bridas, o pasándolos por la madera, de forma que el aspecto general del sistema mejora y los cables quedan más protegidos.

4.3 Montaje

El ensamble de las piezas del afinador es sencillo, ya que el “bastidor” que va a sujetar toda la electrónica se compone de 3 piezas de madera y unos tornillos.

Lo primero que se monta son los motores entre las dos piezas de madera más grandes, se colocan los tornillos de M3x40 y se aprietan hasta que los motores se quedan fijos. Con los motores colocados se pueden probar las piezas que unen el eje de la clavija con el de la guitarra. Sólo hay que desmontar la parte exterior de las clavijas de la guitarra, colocar las piezas en los ejes de las clavijas y meter el eje de los motores por el otro extremo de la pieza:



Figura 4.12 Primera etapa de montaje y prueba de las piezas “unión ejes”

Al intentar montar las piezas de unión de los ejes, resultaron ser “no válidas” debido a que los tornillos no roscaban adecuadamente. Los tornillos tocaban con los ejes antes de haber empezado a roscar y no se podía hacer la presión necesaria para que los ejes giren solidarios. Para solucionar el problema se ha aumentado el diámetro exterior de la pieza a 10 mm, dejando el resto de medidas inalteradas. Se consigue una longitud de roscado de tres milímetros, suficiente para que el prisionero empiece a roscar antes de tocar con el eje. Las nuevas piezas son:



Figura 4.13 Rediseño de unión ejes

Se ha comprobado empíricamente que estas piezas rediseñadas funcionan correctamente.

Después se pueden atornillar el arduino y la PCB de los LEDs y los pulsadores, y pegar las PCBs del amplificador y los drivers a la madera. Con las PCBs colocadas se pueden medir los cables y fabricarlos. Debido a la cantidad de cables a conectar, este es el punto más delicado del montaje. Si algún cable no está en su sitio, se puede estropear el sistema completo.

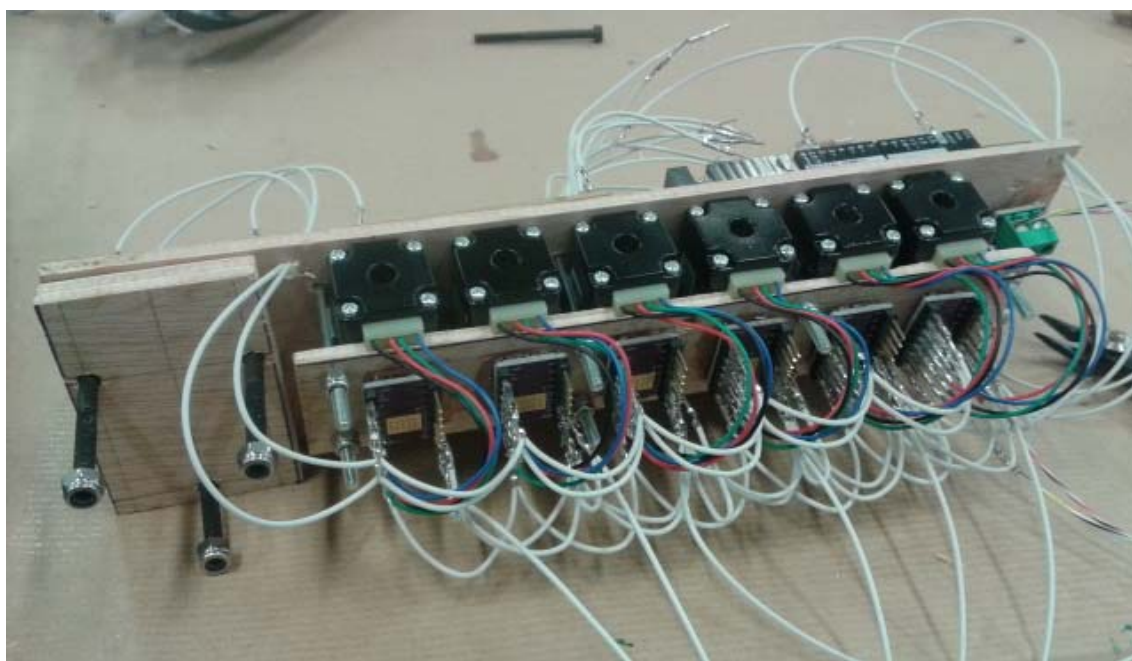


Figura 4.14 Montaje del cableado

Una vez puestos todos los cables, solo hay que conectar el afinador a la guitarra a través de las piezas “unión ejes”, y apretar bien los tornillos para que no se salga el sistema. El afinador después del montaje queda:

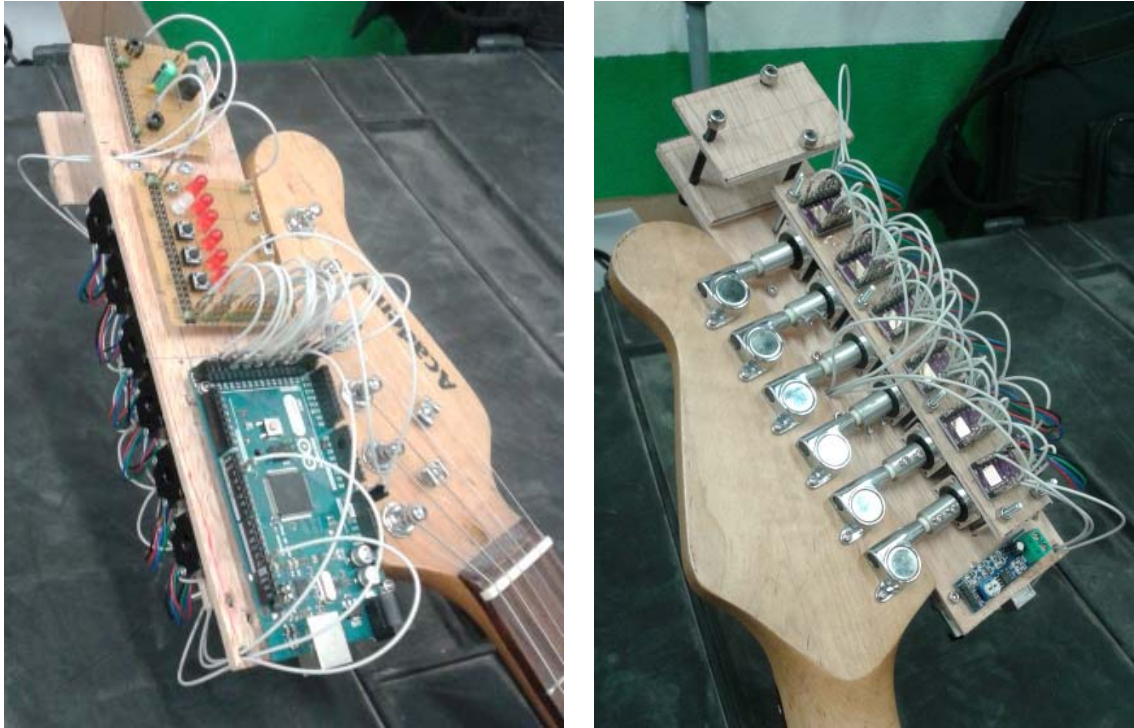


Figura 4.15 Montaje final

4.4 Pruebas

Hay que probar el afinador para determinar si las decisiones tomadas han sido acertadas, y ver si se cumplen los objetivos propuestos.

4.4.1 Prueba de LEDs y pulsadores

La PCB de los pulsadores y los LEDs funciona correctamente. En un principio, los pulsadores presentaban efecto rebote, pero una vez programado y calibrado el delay después de cada lectura del pulsador el efecto desapareció.

Los LEDs se encienden y se apagan correctamente al ir cambiando el modo del programa o de cuerda.

4.4.2 Prueba de los motores

Poniendo el programa en el modo manual y alimentando con una fuente regulada a 18V, se testea cual es la tensión (mecánica) máxima que pueden aplicarle los motores a las cuerdas, para ver si tienen capacidad para afinar la guitarra en afinación estándar.

De estas pruebas se obtienen dos resultados importantes:

- **Los motores no tienen suficiente fuerza como para afinar la guitarra.** Esto quiere decir que se han dimensionado mal y que probablemente no se han tenido en cuenta esfuerzos críticos que intervenían en el proceso, como el rozamiento axial de los ejes contra su carcasa. Estos son los resultados de la prueba:

Cuerda	Afinación conseguida		Afinación deseada	
6ª	Do	66 Hz	Mi	82 Hz
5ª	Mi	82 Hz	La	110 Hz
4ª	La	110 Hz	Re	147 Hz

3ª	Mi	165 Hz	Sol	196 Hz
2ª	La	220 Hz	Si	247 Hz
1ª	Re	294 Hz	Mi	330 Hz

Figura 4.16 Medidas de frecuencia máximas para cada cuerda

El sistema se queda entre uno y cuatro tonos por debajo de la afinación normal. Este fallo es crítico para el sistema, ya que por limitaciones mecánicas no se puede afinar la guitarra para poder tocar con ella. Pero por otro lado, el objetivo del proyecto no era hacer un producto que funcionara a nivel comercial, asique solamente se le va a requerir que **afine**, aunque la afinación no sea la normal. Para una futura iteración habrá que: o cambiar los motores por otros más potentes, o introducir más elementos en la reductora mecánica. Para este sistema la afinación va a ser:

1ª cuerda (Hz)	2ª cuerda (Hz)	3ª cuerda (Hz)	4ª cuerda (Hz)	5ª cuerda (Hz)	6ª cuerda (Hz)
C# - 277.18	A# - 233.1	D# - 155.6	G - 98	E - 82,4	A - 55

Figura 4.17 Nueva afinación objetivo

- Aprovechando la prueba se ha medido el consumo del sistema en cada uno de sus estados:

Interruptor	Arduino	LEDs encendidos	Motor	Consumo
Abierto	Apagado	0	Apagado	0,006 A
Cerrado	Encendido	2	Encendido	0,102 A
Cerrado	Encendido	2	Encendido	2,306 A

Figura 4.18 Consumo del sistema

Cuando un motor está girando el sistema consume 2,3 amperios, a 18V hace una potencia de:

$$P = V \cdot I = 41.4 \text{ W}$$

Ninguna pila de 9V puede trabajar a esa potencia, ya que están optimizadas para 3-7 vatios. Además, en términos de energía, una pila de 300 mAh entregando 2,3 A tardaría aproximadamente 8 minutos en gastarse. Por lo tanto, **no se pueden utilizar pilas**. Hay que pensar en poner una batería.

4.4.3 Prueba del modo automático

La primera prueba del modo automático consiste en medir la precisión y repetitividad de los resultados obtenidos al calcular la frecuencia fundamental del sonido capturado. La precisión como ya se explicó en el punto 3.3.2 depende de la cuerda que se toque y esta prefijada por el programa.

La repetitividad de las muestras se ha visto que no es perfecta, pero es aceptable, aproximadamente el 70% de las medidas son válidas. El problema es que la dispersión de las medidas erróneas es bastante alta. La causa de esta dispersión es el ruido electromagnético, la solución podría ser **augmentar la ganancia del amplificador** para que el sonido capturado tenga mucho más peso en la FFT.

Por otro lado, el algoritmo de control de los motores permite solventar este problema. Tal y como está hecho, la afinación de cada cuerda debería aproximarse mucho a la deseada con

una sola captura correcta de sonido. Por lo tanto, si los motores giran ante una medida errónea, en cuanto haya una medida correcta de la nueva tensión de las cuerdas se corrigen.

4.4.4 Pruebas de control automático de los motores

Para poder configurar el control de los motores, primero se han tenido que tomar medidas de frecuencia en todas las cuerdas para caracterizar la variación de frecuencia al destensar la cuerda un número de pasos determinado. La medida de frecuencias se ha hecho con Matlab, con el hardware y el programa desarrollado para el Experimento 2 del apartado 3.3.2.3 “Algoritmo de obtención de la frecuencia fundamental”. El número de pasos entre medida y medida varía dependiendo de la cuerda, pues la distancia entre notas en hertzios disminuye al disminuir la frecuencia. Para mover los motores se ha hecho un código sencillo en arduino, estos son los resultados:

Medida	1ª cuerda	2ª cuerda	3ª cuerda	4ª cuerda	5ª cuerda	6ª cuerda
1	295,64	249,0	171,1	107,0	89,0	65,8
2	289,66	238,0	167,6	104,2	86,1	64,6
3	281,7	227,0	158,9	100,1	82,6	62,9
4	272,36	216,5	150,1	95,1	79,2	59,0
5	264,2	207,2	142,0	89,0	76,8	56,5
6	256,1	196,7	131,5	83,0	74,5	53,2
7	247,9	186,2	123,4	77,4	72,8	
8	237,4	173,4	114,6	72,2	69,4	
9	228,13	161,8	109,4			
10	217,6	150,2				
11	209,5					
12	197,9					
13	188,5					
Paso	150 ms	100 ms	50 ms	40 ms	30ms	25 ms

Figura 4.19 Frecuencia vs tiempo de giro

Estos datos se van a introducir en arduino para calcular el tiempo de giro (=número de pasos) de los motores para afinarse.

Una vez probado este modo de control, se puede afirmar que es válido, pero depende expresamente de que la medida de la frecuencia fundamental sea correcta.

5. Recursos.

En este apartado se van a evaluar los recursos utilizados, tanto el coste de material como las horas trabajadas para poder corroborar si han cumplido los objetivos y requisitos del proyecto.

5.1 Coste de material:

Reflejan el coste del material que lleva actualmente el prototipo. Se ha dividido entre mecánica y electrónica para evaluar donde está el coste real del prototipo. El apartado “otros costes” corresponde a material para hacer pruebas, material comprado por decisiones erróneas, y a material destruido en el transcurso del proyecto, (PCBs quemadas, cables cortados). Es difícil de contabilizar, por lo que se ha decidido poner una cifra aproximada.

MECÁNICA				
Componente	Cantidad	Precio unitario	Precio total	
Clavijas pack de 6	1	16,8	16,8	
Unión ejes	6	8	48	
Madera	1	3,4	3,4	
Tornillos M3x40 DIN912	11	0,12	1,32	
Tornillos M3x14 DIN912	8	0,1	0,8	
Pasadores M2x3 DIN913	12	0,28	3,36	
Tuercas M3	19	0,13	2,47	
Tornillos M3x50 DIN912	4	0,3	1,2	
		TOTAL	77,35	
ELECTRÓNICA				
Componente	Cantidad	Precio unitario	Precio total	
Arduino Mega 2680	1	34,5	34,5	
Motores	6	23,21	139,26	
Amplificador LM386	1	1,66	1,66	
Driver drv8825	6	2,32	13,92	
PCB Regulador LM7808	1	2,73	2,73	
Pila 9V	2	4,23	8,46	
Conector pila	2	0,97	1,94	
Cable AWG26 (m)	4	0,96	3,84	
Pines macho	50	0,09	4,5	
Pines hembra	50	0,082	4,1	
Termorretractil (m)	1	2,56	2,56	
PCB indicadores	1		0	
	Led rojo	6	0,15	0,9
	Led bicolor	1	0,2	0,2
	Pulsador	3	0,3	0,9
	Resistencia 330	8	0,1	0,8
	Resistencia 4,7K	3	0,1	0,3
	Interruptor	1	0,9	0,9
	Placa de puntos	1	4,5	4,5
		TOTAL	225,97	
OTROS COSTES				
		TOTAL	30	

Sumando los tres costes se obtienen un gasto total de 333,32 euros. Es 30 euros por encima del objetivo fijado al inicio del proyecto. En realidad, el coste de material ha sido menor porque algunos componentes son prestados, (como el Arduino), las horas de trabajo para hacer las 6 piezas de unión de los ejes del motor con el de la clavija han sido gratis (a modo de favor personal, por lo que realmente habrán costado unos 8 euros, en coste de material y herramientas) y como el prototipo se ha montado en una guitarra antigua de baja gama, no ha habido que comprar unas clavijas nuevas.

Se puede considerar entonces que el objetivo de coste de material se ha cumplido.

5.2 Recursos asignados a las tareas.

Para valorar un poco mejor lo que habría costado realmente el prototipo, incluyendo horas de investigación, desarrollo y montaje, se ha hecho una lista de tareas con las horas dedicadas a cada una.

TAREAS	HORAS
Mecánica	93
Calculo tensión cuerdas y par motor	4
Elección reductora	20
Selección de motor y compra	6
Cálculos resistencia reductora	24
Diseño y calculo sistema de anclaje motores	6
Diseño en 3D del sistema	8
Fabricación piezas anclaje de motores en taller	5
Compra y fabricación piezas de madera	5
Compra tornillería	3
Montaje de sistema	12
Electrónica	65
Elección y compra del piezoeléctrico	2
Elección, compra y pruebas del amplificador	20
Elección compra y pruebas del Arduino Mega	12
Diseño compra y fabricación y pruebas PCB LEDs y pulsadores	8
Diseño y fabricación del cableado	14
Elección y compra regulador de tensión	3
Elección, compra y pruebas de drivers del motor	6
Software	70
Caracterización del piezoeléctrico y la vibración de la madera	26
Modo inicio	2
Modo manual	4
Modo automático	32
Pruebas frecuencias variando la tensión de las cuerdas	6
Documentación	69
Memoria	45
Esquemas de cableado/ circuitos/ placement PCB	14
Planos de piezas y dibujos/esquemas explicativos	10
Otros	40
Otros	40

Total	337
--------------	------------

El apartado “otros” tiene en cuenta tiempo improductivo a la espera de recibir el material, tiempo de preparación y limpieza del espacio de trabajo, etc.; y las tareas que no están definidas en la lista pero en las que se ha invertido tiempo.

Si las horas trabajadas se pagaran a precio de becario: 800 euros brutos al mes a jornada completa (8h), el desarrollo y el montaje del prototipo habrían costado, suponiendo que el mes tiene 22 días laborables:

$$337 \cdot \frac{800}{22 \cdot 8} = 1531,81 \text{ euros}$$

Sumándole el precio de los componentes:

$$1531,81 + 333,32 = 1865,13 \text{ euros}$$

Tanto el número de horas como el coste total son cantidades totalmente aceptables para el desarrollo de un prototipo, y desde un punto de vista puramente económico animan a seguir trabajando en el proyecto para mejorarlo.

6. Conclusiones:

En vista de los objetivos y requisitos y planteados al inicio del proyecto y los recursos con los que se contaba (un presupuesto de 300 euros y un ingeniero mecánico), se podría decir que el resultado ha sido satisfactorio. Sin perder de vista que el sistema construido está muy lejos de acercarse a un producto comercializable, es un muy buen comienzo para la siguiente etapa del proyecto. Esta nueva etapa consistiría en construir un segundo prototipo partiendo del actual, que mantuviera las cualidades del primero y mejorara hasta donde se pudiera los defectos. Para ello, se va a evaluar el sistema de la forma más crítica posible, tratando de encontrar los puntos fuertes y las debilidades, para luego plantear las posibles mejoras a llevar a cabo en la siguiente fase del proyecto.

6.1 Evaluación del sistema.

En general todo el sistema es barato y sencillo, como se pretendía, pero, es un prototipo deficiente en algunos aspectos. Hay tres puntos críticos que habría que solucionar antes de pensar en una nueva etapa:

1. Los motores tienen que tener la capacidad de tensar la cuerda hasta por encima de la tensión que corresponda a la nota afinada.
2. La alimentación del sistema tiene que garantizar una autonomía de al menos una hora funcionando y ser independiente de la red eléctrica.
3. La confiabilidad de los resultados obtenidos al calcular la frecuencia fundamental tiene que aproximarse al 100%.

Además de estos tres puntos, existen una serie de deficiencias secundarias:

- El conjunto es demasiado grande, aunque no estorbe cuando se toca. Sobresale de la guitarra y hace imposible que se pueda guardar en un estuche convencional.
- Como no se han puesto carcasas, el sistema no está protegido contra polvo, agua, golpes y/o caídas, por lo que se prevé una vida corta para el prototipo.
- Las piezas que unen las clavijas de la guitarra con los motores únicamente sirven para que los ejes giren juntos, si se tira de un motor en sentido axial al su eje, el sistema se puede desacoplar.
- Los motores son demasiado grandes, aunque estén escogidos acorde a la tensión de las cuerdas y la reductora. Aun así, la elección de motores eléctricos tipo paso a paso es acertada.
- Fabricar piezas en madera es muy cómodo porque se mecaniza con mucha facilidad pero no es apto para un producto final.
- La plataforma Arduino tiene muchas ventajas, como a la accesibilidad económica, sencillez de uso y la cantidad de código que se puede encontrar en la red, pero no está optimizado para la aplicación en la que se usa, está falto de recursos en algunos aspectos (por ejemplo, el ADC solo admite tensiones positivas) y sobrado para otros (56 entradas y salidas configurables diferentes, de las cuales se usan algo más de la mitad).
- La amplificación para el piezoeléctrico es muy pobre, si se usaran más etapas de amplificación, se evitarían problemas de ruidos al buscar la frecuencia del sonido,

porque los máximos sobresaldrían más en la FFT. Aunque haga falta amplificarlo, utilizar un piezoeléctrico en vez de un micrófono es una decisión correcta a mantener.

- No se han usado conectores para los cables, en parte porque las PCBs se han comprado ya montadas y soldadas. Pero obliga a soldar cada cable para evitar que se suelte, impidiendo cambiar los componentes por separado en caso de que se estropeen.
- Las pilas, sean o no recargables dan poca potencia y se descargan muy rápido si se está utilizando el sistema de manera continuada.
- La librería utilizada para calcular la transformada de Fourier toma un número de muestras insuficiente para la precisión requerida en un producto final.
- Afinar una cuerda cada vez hace que el proceso de afinación sea muy lento, además de poco útil, porque si el usuario se equivoca de cuerda el programa la afinará mal.
- El programa solo permite un tipo de afinación, la estándar, es una pérdida de funcionalidad en la propia guitarra y para el músico, que dependiendo de lo que toque querrá variar la afinación de algunas o todas las cuerdas.
- El sistema tiene una estética nula.

6.2 Mejoras al prototipo actual.

Antes de plantear las diferentes soluciones a los problemas propuestos, hay puntualizar que para poder llevarlas a cabo haría falta aumentar los recursos destinados al proyecto. Primero, parte de las decisiones tomadas que se han considerado como “deficiencias” en el punto anterior, se deben a una limitación de presupuesto, si se quiere mejorar el prototipo hay que encontrar una fuente externa de financiación. Y segundo, las deficiencias a nivel de programación y electrónica se debe a no haber contado con expertos en esas ramas, por lo que para la siguiente etapa, convendría involucrar en el proyecto a algún ingeniero electrónico y/o de telecomunicaciones, para mejorar la programación y los circuitos. Si se cuenta con tres ingenieros (mecánico, electrónico y telecomunicaciones) y un presupuesto adecuado, se pueden llevar a cabo las siguientes mejoras:

- Sustituir las clavijas de la guitarra por otras de diseño particular que incorporen una reductora compuesta por un tren de engranajes, y que incluya la parte de la clavija que sirve para girarla con la mano. Debería tener una reducción de par tal, que el tamaño de los motores disminuya al menos a la mitad, pero con la potencia suficiente como para que la afinación no esté limitada por los motores. El motor y la reductora se podrían introducir en una carcasa metálica individual (para cada conjunto reductora-motor) de forma que solo salieran 4 cables de ella, uno para cada fase. De esta forma se reduciría el tamaño del sistema, y se eliminarían piezas susceptibles de fallar como la unión de los ejes del motor y el sinfín.
- Diseñar una carcasa que incluya todos los componentes. Se podría prescindir de las piezas de madera y dándole al sistema robustez y una gran mejora estética. Con las nuevas tecnologías de impresión 3D, como la sinterización laser, se pueden conseguir pequeñas tiradas de piezas de una calidad notable a un bajo precio, ideal para hacer prototipos.

- Habría que estudiar si el afinador podría posicionarse sobre la superficie de la pala de la guitarra, en vez de sobresaliendo. Así se podrían evitar los problemas de almacenaje actuales.

Para estas tres tareas se asignaría a un ingeniero mecánico.

- Diseñar un circuito amplificador de una ganancia mayor al actual, para ayudar a mejorar la detección de la frecuencia fundamental del sonido.
- Rediseñar e incluir toda la electrónica en una sola PCB o varias apiladas utilizando la tecnología SMT (montaje superficial) para disminuir su tamaño. Esta/s PCB incluiría el mismo tipo de componentes del prototipo: un regulador de voltaje, las etapas de amplificación necesarias, el microcontrolador y los drivers de los motores, pero cada elemento estaría estudiado y optimizado para su función en el sistema. Esto mejoraría el funcionamiento del afinador y a la vez reduciría su tamaño.
- Incluir conectores para los cables en la PCB. Esto hace el sistema más modular, y permitiría cambiar una parte estropeada con solo desconectar el cable.
- Debido al problema de consumo actual, que no permite usar pilas, habría que utilizar baterías. De esta forma se aumentaría además la autonomía, aunque conlleve el diseño de un cargador y un circuito de protección.

Estas cuatro tareas se asignarían al ingeniero electrónico.

- Desarrollar o buscar un algoritmo para hacer la transformada rápida de Fourier que permita procesar más muestras, de forma aumente la precisión de la afinación.
- Mejorar el programa para que se puedan tocar todas las cuerdas a la vez, y que el programa sea capaz de identificarlas y afinarlas simultáneamente.
- Introducir un banco de afinaciones en el programa y algún otro pulsador e indicador LED más, para que el usuario pueda elegir y cambiar la afinación que necesita.

Y estas tres últimas al ingeniero de telecomunicaciones.

6.4 Mejoras diferenciadoras

Ciertamente, aunque todas estas mejoras son lógicas y necesarias, acercan el posible producto al diseñado por Tronical. No tiene sentido invertir recursos en desarrollar un producto igual, ya que Tronical ofrece calidad a un precio relativamente bajo con el que difícilmente se puede competir. Es interesante y necesario plantear una serie de mejoras diferenciadoras que hagan que el usuario perciba el producto de una forma diferente al de Tronical. Ya que difícilmente se podría competir en precios, se justificaría un precio más elevado con un aumento de funcionalidad:

- Se podría diseñar la mecánica del sistema de forma que el afinador automático en vez de estar acoplado siempre a la misma guitarra se pudiera enganchar y desenganchar de las clavijas y sirviera para cualquier guitarra. De esta forma la guitarra no tendría ningún tipo de hardware acoplado, y cuando se desafinara se le colocaría el afinador, se tocarían las cuerdas, se afinarían automáticamente y se quitaría el aparato para seguir tocando.

- Añadir funcionalidad al sistema, por ejemplo, se podría colocar una pantalla LCD y ponerle un modo de afinación manual, que al tocar la cuerda mostrara la nota que estuviera sonando en la pantalla y se pudiera afinar con la mano. O también, estudiar exhaustivamente la variación de la frecuencia de cada cuerda en función de los pasos del motor y programar cambios de afinación en bucle abierto, es decir, al pulsar un botón y sin recoger muestras de sonido, los motores podrían cambiar la afinación de la guitarra, incluso sin dejar de tocar.
- Incluir un elemento más en el diseño que hiciera sonar las cuerdas también de forma automática, tipo eBow, que es un dispositivo que mediante unas bobinas induce vibración en las cuerdas. De esta manera, no sería necesario tocar las cuerdas para afinarlas y el proceso de afinación se volvería completamente automático.
- Aplicar lo aprendido para adaptar el afinador a otro tipo de instrumentos de cuerda que sean propensos a desafinarse con facilidad y cuya afinación sea tediosa. Como un violín, que es muy sensible a los cambios de temperatura y se desafina con muchísima facilidad; o, un arpa de orquesta, con sus 47 cuerdas afinadas una a una. Supondría ocupar un nicho de mercado totalmente nuevo. Si bien hay que mencionar que el número de arpistas en comparación con el de guitarristas es mucho menor, habría que hacer un estudio de mercado para ver si merece la pena invertir recursos en desarrollar un producto de estas características.

Si se optara seguir adelante y empezar una nueva fase del proyecto, habría que decidir cuál de estas cuatro opciones es mejor y añadirla a la lista general de mejoras. Si ninguna es viable, posiblemente no compense continuar.

Anexos

Anexo A: Propiedades mecánicas de algunos materiales

Acero inoxidable:

	Grado	Producto ¹⁾	Espesor máximo (mm)	Mínima resistencia ²⁾ correspondiente al 0.2% (N/mm ²)	Resistencia última a tracción (N/mm ²)	Alargamiento de rotura (%)
Aceros inoxidables austeníticos básicos de cromo y níquel	1.4301	C	8	230	540 – 750	45 ⁽³⁾
		H	13,5	210	520 – 720	45 ⁽³⁾
		P	75	210	520 – 720	45
Aceros inoxidables austeníticos de molibdeno, cromo y níquel	1.4307	C	8	220	520 – 700	45
		H	13,5	200	520 – 700	45
		P	75	200	500 – 700	45
Aceros inoxidables austeníticos de molibdeno, cromo y níquel	1.4401	C	8	240	530 – 680	40
		H	13,5	220	530 – 680	40
		P	75	220	520 – 670	45
Aceros inoxidables austeníticos estabilizados	1.4541	C	8	220	520 – 720	40
		H	13,5	200	520 – 720	40
		P	75	200	500 – 700	40
Aceros inoxidables austeníticos bajos en carbono, altos en nitrógeno	1.4571	C	8	240	540 – 690	40
		H	13,5	220	540 – 690	40
		P	75	220	520 – 670	40
Aceros inoxidables dúplex	1.4318	C	8	350	650 – 850	35
		H	13,5	330	650 – 850	35
		P	75	330	630 – 830	45
Aceros inoxidables dúplex	1.4362	C	8	450	650 – 850	20
		H	13,5	400	650 – 850	20
		P	75	400	630 – 800	25
	1.4462	C	8	500	700 – 950	20
		H	13,5	460	700 – 950	25
		P	75	460	640 – 840	25

Notas:

1) C=fleje laminado en frío, H=fleje laminado en caliente, P=chapa laminada en caliente

2) Propiedades transversales

3) Para material más estirado, los valores mínimos son un 5% más bajos

Figura A.1 Propiedades mecánicas de los aceros inoxidables usuales. [22]

Bronce:

Número UNS	Nombre de la aleación	Forma	Revenido	Resistencia de fluencia MPa	Resistencia a la tensión MPa	Elongación en 50 mm %	Dureza Rockwell H_R
C17000	Berilio	Varilla	Duro	515	790	5	98B
		Varilla	Suave	170	415	50	77B
		Lámina	Duro	1000	1240	2	...
C21000	Latón de dorar	Lámina	Duro	345	385	5	64B
		Lámina	Suave	70	235	45	46F
C22000	Bronce comercial	Lámina	Suave	370	420	5	70B
		Lámina	Suave	70	255	45	53F
		Varilla	Duro	380	415	20	60B
		Varilla	Suave	70	275	50	55F
C23000	Latón rojo	Lámina	Duro	395	480	5	77B
		Lámina	Suave	85	275	47	59F
		Varilla	Duro	360	395	23	75B
		Varilla	Suave	70	275	55	55F
C26000	Latón para cartuchos	Lámina	Duro	435	525	8	82B
		Lámina	Suave	105	325	62	64F
		Varilla	Duro	360	480	30	80B
		Varilla	Suave	110	330	65	65F
C27000	Latón amarillo	Lámina	Duro	415	510	8	80B
		Lámina	Suave	105	325	62	64F
		Varilla	Duro	310	415	25	80B
		Varilla	Suave	110	330	65	65F
C28000	Metal Muntz	Lámina	Duro	415	550	10	85B
		Lámina	Suave	145	370	45	80F
		Varilla	Duro	380	515	20	80B
		Varilla	Suave	145	370	50	80F
		Tubo	Duro	380	510	10	80B
C33000	Latón bajo plomo	Tubo	Duro	415	515	7	80B
		Tubo	Suave	105	325	60	64F
C33200	Latón de alto plomo	Lámina	Duro	415	510	7	80B
		Lámina	Suave	115	340	52	68F
C46200	Latón naval	Lámina	Duro	480	620	5	90B
		Varilla	Duro	365	515	20	82B
		Tubo	Duro	455	605	18	95B

Figura A.2 Propiedades típicas de algunas aleaciones con base cobre. [23]

Aluminio:

Nombre	Resistencia a la fluencia	Resistencia máxima	Elongación	Modulo de elasticidad	Densidad	Norma
3003 H14 (UNS A93003)	145 MPa (21 KSI)	152 MPa (22 KSI)	8% (en 50mm, espesor 1.6mm))	69 GPa (10000 KSI)	2.73 g/cm ³	ASTM B-209
6061 T651 (UNS A96061)	255 MPa (37 KSI)	290 MPa (42 KSI)	12% (en 50mm, espesor 1.6mm))	69 GPa (10000 KSI)	2.7 g/cm ³	ASTM B 221
7075 T651 (UNS A97075)	427 MPa (62 KSI)	503 MPa (73 KSI)	8% (en 50mm)	72 GPa (10400 KSI)	2.81 g/cm ³	ASTM B211

Figura A.3 Propiedades mecánicas de algunas aleaciones de aluminio [24]

Anexo B: Código del afinador en arduino y matlab.

B.1 Código del sistema:

```
//incluir librerías
#define LOG_OUT 1
#define FFT_N 256
#include <FFT.h>
//asignacion de pines
int led6=30;
int led5=33;
int led4=34;
int led3=37;
int led2=38;
int led1=41;
int ledautomatico=42;
int ledmanual=45;
int subir=29;
int bajar=25;
int cambiomodo=26;
int direccion=46;
int motoron[]={48,49,50,51,52,53};
int stp=2;
int frecuerda[2];
//variables dependientes de las cuerdas
int cortesuperior[]={300,250,170,110,90,65};
int corteinferior[]={190,150,110,70,70,50};
int cuerda=5;
int LEDS[]={led1,led2,led3,led4,led5,led6,ledautomatico,ledmanual};
int microdelay[]={860,860,1550,1550,3100,3100};
double
F1[]={295.6,289.7,281.7,272.4,264.2,256.1,247.9,237.4,228.1,217.6,209.
5,197.9,188.5};
double F2[]={249,238,227,216.5,207.2,196.7,186.2,173.4,161.8,150.2}
double F3[]={171.1,167.6,158.9,150.1,142,131.5,123.4,114.6,109.4}
double F4[]={107,104.2,100.1,95.1,89,83,77.4,72.2}
double F5[]={89,86.1,82.6,79.2,76.8,74.5,72.8,69.4}
double F6[]={65.8,64.6,62.9,59,56.5,53.2}
double Fobj[]={277.18,233.1,155.6,98,82.4,55};
int delaymotor[]={150,100,50,40,30,25}
//variables de estado
bool modoinicio=true;
bool modoautomatico=false;
bool modomanual=false;
bool Vref=false;
//variables frecuencia
int frec;
void setup() {
pinMode(led1,OUTPUT);
pinMode(led2,OUTPUT);
pinMode(led3,OUTPUT);
pinMode(led4,OUTPUT);
pinMode(led5,OUTPUT);
pinMode(led6,OUTPUT);
pinMode(ledautomatico,OUTPUT);
pinMode(ledmanual,OUTPUT);
pinMode(subir,INPUT);
pinMode(bajar,INPUT);
pinMode(cambiomodo,INPUT);
Serial.begin(9600);
```

```

}
//-----
//programa principal
void loop() {
if (modoinicio){
    inicio();
}
if(modoautomatico){
//se comprueba si se han pulsado los pulsadores de cambio de modo y/o
cuerda
    cambiodemodo();
    subirbajarcuerda();
//se comprueba si el usuario ha tocado alguna cuerda
    if(analogRead(A0)>600){
//se identifica la frecuencia fundamental
        frec=captura(cuerda);
        frecuencia[1]=frec;
        frecuencia[2]=cuerda;
//se pone en marcha el motor
        afinar(frecuencia);
        Serial.println(cuerda);
    }
}
if(modomanual){
//se comprueba si se han pulsado los pulsadores de cambio de modo y/o
cuerda
    cambiodemodo();
    subirotensarcuerda();
    bajarodestensarcuerda();
}
}
//-----
//Funcion "INICIO" secuencia de encendido y apagado de los LEDs
void inicio() {
int i;
for(i=5;i>-1;i--){
    digitalWrite(LEDs[i],HIGH);
    delay(300);
}
digitalWrite(LEDs[6],HIGH);
delay(300);
digitalWrite(LEDs[6],LOW);
digitalWrite(LEDs[7],HIGH);
delay(300);
apagarleds();
for(i=0;i<8;i++){
    digitalWrite(LEDs[i],HIGH);
}
delay(300);
apagarleds();
for(i=0;i<8;i++){
    digitalWrite(LEDs[i],HIGH);
}
delay(300);
apagarleds();
digitalWrite(led6,HIGH);
digitalWrite(ledautomatico,HIGH);
modoinicio=false;
modoautomatico=true;
}

```

```

//-----
//Funcion "APAGAR LEDS" apaga todos LEDS
void apagarleds () {
    int i;
    for(i=0;i<8;i++){
        digitalWrite(LEDs[i],LOW);
    }
    delay(300);
}
//-----
//Funcion "CAMBIO DE MODO" Cambia de modo manual a automático y viceve
rsa
void cambiodemodo() {
if(modoautomatico) {
    if (digitalRead(cambiomodo)) {
        modoautomatico=false;
        modomanual=true;
        apagarleds();
        cuerda=5;
        digitalWrite(LEDs[cuerda],HIGH);
        digitalWrite(ledmanual,HIGH);
        delay(500);
    }
}
if(modomanual) {
    if (digitalRead(cambiomodo)) {
        modomanual=false;
        modoautomatico=true;
        apagarleds();
        cuerda=5;
        digitalWrite(LEDs[cuerda],HIGH);
        digitalWrite(ledautomatico,HIGH);
        delay(1000);
    }
}
}
//-----
//Funcion "SUBIR O TENSAR CUERDA" del modo manual:
void subirotensorcuerda() {
unsigned long tstart;
unsigned long tend;
if(digitalRead(subir)) {
    tstart=millis();
    while (digitalRead(subir)) {
        tend=millis();
//Si se pulsa el pulsador mas de 500 ms, el motor empieza a moverse
        if(tend-tstart>500) {
            digitalWrite(motoron[cuerda],HIGH);
            digitalWrite(direccion,LOW);
            analogWrite(stp,150);
            delay(50);
        }
    }
    digitalWrite(motoron[cuerda],LOW);
//Si el tiempo es menor, solamente se cambia de cuerda
    if(tend-tstart<500) {
        digitalWrite(LEDs[cuerda],LOW);
        if (cuerda>0) {

```

```

        cuerda=cuerda-1;
    }
    else{
        cuerda=5;
    }
    digitalWrite (LEDS [cuerda], HIGH);
    delay (200);
}
}
}
//-----
//Funcion "BAJAR O DESTENSAR CUERDA" del modo manual:
void bajarodestensarcuerda () {
    unsigned long tstart;
    unsigned long tend;
    if (digitalRead (bajar)) {
        tstart=millis ();
        while (digitalRead (bajar)) {
            tend=millis ();
//Si se pulsa el pulsador más de 500 ms, el motor empieza a moverse

            if (tend-tstart>500) {
                digitalWrite (motoron [cuerda], HIGH);
                digitalWrite (direccion, HIGH);
                analogWrite (stp, 150);
                delay (50);
            }
        }
        digitalWrite (motoron [cuerda], LOW);
//Si el tiempo es menor, solamente se cambia de cuerda
        if (tend-tstart<500) {
            digitalWrite (LEDS [cuerda], LOW);
            if (cuerda<5) {
                cuerda=cuerda+1;
            }
            else {
                cuerda=0;
            }
            digitalWrite (LEDS [cuerda], HIGH);
            delay (200);
        }
    }
}
//-----
//Funcion "SUBIR Y BAJAR CUERDA" del modo automatico: Permite cambiar
la cuerda a afinar con los pulsadores
void subirbajarcuerda () {
    if (digitalRead (bajar)) {
        digitalWrite (LEDS [cuerda], LOW);
        if (cuerda<5) {
            cuerda=cuerda+1;
        }
        else {
            cuerda=0;
        }
        digitalWrite (LEDS [cuerda], HIGH);
        delay (200);
    }

    if (digitalRead (subir)) {

```

```

    digitalWrite(LEDs[cuerda],LOW);
    if (cuerda>0){
        cuerda=cuerda-1;
    }
    else{
        cuerda=5;
    }
    digitalWrite(LEDs[cuerda],HIGH);
    delay(200);
}
}
//-----
//Funcion "CAPTURA" del modo automatico: Calcula la frecuencia fundame
ntal del sonido
int captura(int c){
    unsigned long tstart, tend;
    int i;
    int fmax;
    int frec;
    int media=0;
    double sep;
//se adquieren 512 muestras, (solo son necesarias las 256 pares, que c
orresponden a numeros reales)
    tstart=micros();
    for (int i = 0 ; i < 512 ; i += 2) {
        int b=analogRead(A0);
        fft_input[i]=b;//-media;
        fft_input[i+1]=0;
        delayMicroseconds(microdelay[cuerda]);
    }
    tend=micros();
//Se hacen transformaciones necesarias para aumentar la velocidad del
algoritmo
    fft_window();
    fft_reorder();
    fft_run();
    fft_mag_log();
//se calcula la resolución de la transformada
    sep=1000000/(tend-tstart);
//se filtra por bandas de frecuencia en función de la cuerda que se ha
ya tocado
    for(i=0;i<(corteinferior[c]/sep);i++){
        fft_log_out[i]=0;
    }
    for(i=(cortesuperior[c]/sep);i<128;i++){
        fft_log_out[i]=0;
    }
//se busca el máximo de la transformada y se devuelve al programa prin
cipal
    fmax=maximo(fft_log_out,128);
    frec=fmax*sep;
    Serial.println(frec);
    return(frec);
}
//-----
//Funcion "AFINAR" del modo automatico: Calcula el tiempo que deben mo
verse los motores para alcanzar la frecuencia objetivo
void afinar(int fc[2]){
    int i=0;
    int f=fc[1];
    int c=fc[2];

```



```

double t;
//Interpolacion lineal entre los datos de frecuencias calculados
if (c==0){
    while (F1[i]>Fobj[c]){
        i++;
    }
    double tdelay1=(i-1)+(Fobj[c]-F1[i-1])/(F1[i]-F1[i-1]);
    i=0;
    while (F1[i]>f){
        i++;
    }
    double tdelay2=(i-1)+(f-F1[i-1])/(F1[i]-F1[i-1]);
//Calculo del tiempo de giro de los motores
    t=(tdelay2-tdelay1)*delaymotor[c];
    digitalWrite(motoron[c],HIGH);
    if (t>0){
        digitalWrite(direccion,HIGH);
    }
    else {
        t=-t;
        digitalWrite(direccion,LOW);
    }
    analogWrite(stp,150);
    delay(t);
    digitalWrite(motoron[c],LOW);
    delay(200);
    Serial.println(t);
}
}

//-----
//Funcion "MAXIMO" del modo automatico: Calcula la posicion del maximo
de la transformada
int maximo (uint8_t arr[], int n)
{
    int m = 0;
    int val = 0;
    for (int i = 0; i < n; i++)
    {
        if (arr[i] > val)
        {
            m = i;
            val = arr[i];
        }
    }
    return m;
}
}

```

B.2 Código afinador Matlab

```
%Declaración e inicialización de objetos
while (true)
F= dsp.AudioRecorder('DeviceName','Entrada de línea (2- Scarlett 2i2
USB)', 'NumChannels',1,'SampleRate',44100);
AFW = dsp.AudioFileWriter('muestra.wav','FileFormat',
'WAV','DataType','single');
tic;
%Captura de sonido en un archivo .wav
while toc < 1,
    audioIn = step(F);
    step(AFW,audioIn);
end
release(F);
release(AFW);
%Muestrear el archivo .wav
[V,Fs] = wavread('muestra');
%Calcular la frecuencia fundamental
Frec=Frecuencia(V,Fs);
[N Fprox]=Nota(Frec);
%Mostrar por pantalla la frecuencia obtenida y la nota más cercana con
su frecuencia
disp(Frec)
disp(Fprox)
disp(N)
disp(' _____ ');
end

%-----

function [ FREC ] = Frecuencia( Y,Fs )%introduce .WAV para analizarlo
con la FFT
Y=Y'; %coloca la muestra de audio en un vector horizontal.
Y= Y(1,1:length(Y)); %elimina uno de los dos canales de audio.
Fourier = abs(fft(Y)); %transformada de Fourier.
L=length(Fourier);
Fmax = max(Fourier);%busco la amplitud maxima, como la muestra dura
un segundo, la frecuencia de muestreo es Fs, la separacion entre los
términos de la transformada es Fs/N (donde N es el numero de muestras
de la señal no transformada) si Fs=N hay un Hz de distancia entre cada
muestra.
i=1; %inicializo el contador.
while Fmax ~= Fourier(i)
    i=i+1;
end
FREC=i*Fs/L;%Consigo la frecuencia fundamental del sonido, depende
como se ha explicado de la separacion entre muestras y la frecuencia
de muestreo, por eso, se multiplica el indice "i" para que el
resultado salga en Hz
end

%-----

function [ N,Fprox ] = Nota( Frecuencia ) %con una frecuencia te dice
la nota
Notas=['A ','A#','B ','C ','C#','D ','D#','E ','F ','F#','G ','G#'];
%vector con el nombre de las notas y matriz de frecuencias de notas
Freqs=[27.500 2*27.500 4*27.500 8*27.500 16*27.500 32*27.500
64*27.500;...
```

```

        29.135 2*29.135 4*29.135 8*29.135 16*29.135 32*29.135
64*29.135;...
        30.868 2*30.868 4*30.868 8*30.868 16*30.868 32*30.868
64*30.868;...
        32.703 2*32.703 4*32.703 8*32.703 16*32.703 32*32.703
64*32.703;...
        34.648 2*34.648 4*34.648 8*34.648 16*34.648 32*34.648
64*34.648;...
        36.708 2*36.708 4*36.708 8*36.708 16*36.708 32*36.708
64*36.708;...
        38.891 2*38.891 4*38.891 8*38.891 16*38.891 32*38.891
64*38.891;...
        41.203 2*41.203 4*41.203 8*41.203 16*41.203 32*41.203
64*41.203;...
        43.654 2*43.654 4*43.654 8*43.654 16*43.654 32*43.654
64*43.654;...
        46.249 2*46.249 4*46.249 8*46.249 16*46.249 32*46.249
64*46.249;...
        48.999 2*48.999 4*48.999 8*48.999 16*48.999 32*48.999
64*48.999;...
        51.913 2*51.913 4*51.913 8*51.913 16*51.913 32*51.913
64*51.913];
i=1;
j=1;
while Frecs(i,j) <=Frecuencia && j<7 %bucles para encontrar la
frecuencia mas cercana
    while Frecs(i,j) <=Frecuencia && i<12

        i=i+1;
        k=i;
    end
    l=j;
    j=j+1;
    i=1;
end

if Frecuencia>Frecs(k,l) %problemas de consistencia del programa
    k=1;
    l=l+1;
end
if Frecuencia==Frecs(k,l)
    N=Notas(k);
elseif Frecuencia < Frecs(k,l)
    if k==1 %mas problemas de consistencia
        a=13;
        b=l-1;
    else
        a=k;
        b=l;
    end
    %hay que calcular la nota mas proxima a la frecuencia calculada
    if Frecuencia >= (Frecs(k,l)+Frecs(a-1,b))/2
        N=Notas(k,1:2);
        Fprox=Frecs(k,l);
    else
        N=Notas(a-1,1:2);
        Fprox=Frecs(a-1,l);
    end
end
end
end

```

Bibliografía

- [1] <http://axcenttuning.com/systemfeatures.html> Buscado: 26/05/2015
- [2] <http://www.gibson.com/robotguitar/guitar.html> Buscado: 26/05/2015
- [3] <http://www.tronical.com/about-tronical/z> Buscado: 26/05/2015
- [4] <http://peavey.com/products/at200/> Buscado: 26/05/2015
- [5] <http://line6.com/jtv-69-us/> Buscado: 26/05/2015
- [6] <http://aprendoguitarra.com/tecnica/%c2%bfque-cuerdas-utilizo-medidas-de-cuerdas.php>
Buscado: 05/05/2015
- [7] American Lutherie No 2 Junio 1985
- [8] D'Addario, Catalog Supplement/String Tension Specifications
- [9] <http://www.thomann.de/es/index.html> Buscado: 08/05/2015
- [10] Richard G. Budynas, J. Keith Nisbett, Diseño en ingeniería mecánica de Shigley 8ª edición.
Pag 678.
- [11] Richard G. Budynas, J. Keith Nisbett, Diseño en ingeniería mecánica de Shigley 8ª edición.
Pag 695.
- [12] Imagen suministrada por el fabricante, Stepper Online
- [13] ANSI/AGMA 6034-B92
- [14] Richard G. Budynas, J. Keith Nisbett, Diseño en ingeniería mecánica de Shigley 8ª edición.
Pag 1008
- [15] Richard G. Budynas, J. Keith Nisbett, Diseño en ingeniería mecánica de Shigley 8ª edición.
Pag 287
- [16] Richard G. Budynas, J. Keith Nisbett, Diseño en ingeniería mecánica de Shigley 8ª edición.
Pag 288
- [17] Richard G. Budynas, J. Keith Nisbett, Diseño en ingeniería mecánica de Shigley 8ª edición.
Pag 280
- [18] Richard G. Budynas, J. Keith Nisbett, Diseño en ingeniería mecánica de Shigley 8ª edición.
Pag 283
- [19] <http://www.todorobot.com.ar/tutorial-sobre-motores-paso-a-paso-stepper-motors/>
Buscado: 10/06/2015
- [20] <http://zone.ni.com/reference/en-XX/help/370051V-01/cvi/libref/analysisconcepts/aliasing/> Buscado: 18/06/2015

- [21] <http://asterion.almadark.com/2010/05/31/calibre-de-conductores-y-su-amperaje-maximo/> Buscado: 25/05/2015
- [22] Norma UNE-EN 10088-2
- [23] <http://ingemecanica.com/tutorialsemanal/tutorialn150.html> Buscado: 25/06/2015
- [24] <http://www.sumiteccr.com/Aplicaciones/Articulos/pdfs/AA%206061.pdf> Buscado: 25/06/2015
- [25] Earle Buckingham, Desig of worm and spiral gears, Industrial press, 1981
- [26] De C. J. Noll y C. Lipson, "Allowable Working Stresses", en Society for Experimental Stress Analysis, vol. 3. núm. 2, 1946, p. 29. Reproducida por O. J. Horger (ed.), Metals Engineering Design ASME Handbook, McGraw-Hill, Nueva York. Copyright©1953 por The McGraw-Hill Companies, Inc.
- [27] E. B. Haugen y P. H. Wirsching, "Probabilistic Design", en Machine Design, vol. 47, núm. 12, 1975, pp. 10-14.
- [28] R. E. Peterson, Stress Concentration Factors, Wiley, Nueva York, 1974, pp. 148, 244.