

RECIBIDO: 29 DE MARZO 2024 REVISADO: 3 DE ABRIL 2024 ACEPTADO: 5 DE ABRIL 2024

CURVAS INTERACTIVAS DE LISSAJOUS DESARROLLADAS EN OCTAVE Y ARDUINO

INTERACTIVE LISSAJOUS CURVES DEVELOPED IN OCTAVE
AND ARDUINO

Javier Díaz Sánchez

Benemérita Universidad Autónoma de Puebla, ULC
javier.diazsa@correo.buap.mx

ORCID: 0000-0002-5851-9636

RESUMEN

El aprendizaje de las matemáticas ante el nuevo entorno digital, invita a proponer estrategias didácticas que permitan desarrollar experiencias de aprendizaje significativo, en este trabajo se expone una propuesta orientada para su aplicación y desarrollo en las áreas de matemáticas e informática, para el Nivel Medio Superior (BUAP) durante el sexto semestre del Plan 07. El proyecto trata sobre el trazado gráfico de curvas de Lissajous a través de una interface entre el software matemático Octave y un prototipo basado en una placa Arduino, teniendo este último como función la recopilación de datos externos, generados a partir de la información recopilada por dos palancas de mando (joystick), las cuales permiten la variación de las gráficas a través de su movimiento, lo que modifica directamente los parámetros de la ecuación parametrizada de Lissajous; permitiendo una interacción (hombre-máquina) que integra conocimiento transversal y que contribuye a la tendencia STEM en la educación, así como la aportación documental sobre interfaces electrónicas e informáticas con software libre en el desarrollo de prácticas de laboratorio, que amplía el bagaje de la cultural digital y fortalece el perfil de egreso.

Palabras claves: Informática educativa, Matemáticas, Octave, Aplicación informática, Software de código abierto.

ABSTRACT

The learning of mathematics in the new digital environment invites us to propose didactic strategies that allow the development of meaningful learning experiences. This work presents a proposal aimed at its application and development in the areas of mathematics and computer science, for the Higher Secondary Level (BUAP) during the sixth semester of Plan 07. The project deals with the graphical tracing of Lissajous curves through an interface between the mathematical software Octave and a prototype based on an Arduino board, the latter having data collection as its function. external, generated from the information collected by two joysticks, which allow the variation of the graphs through their movement, which directly modifies the parameters of the parameterized Lissajous equation; allowing an interaction (man-machine) that integrates transversal knowledge and that contributes to the STEM trend in education, as well as the documentary contribution on electronic and computer interfaces with free software in the development of laboratory practices, which expands the background of the digital culture and strengthens the graduation profile.

Key Words: Educational computing, Mathematics, Octave, Computer application, Open source software.

INTRODUCCIÓN

Desde tiempos remotos, los grandes pensadores matemáticos siempre han querido reducir un fenómeno físico o social a una simple ecuación o algoritmo, esto con el fin de calcular y predecir los sucesos que afectan al entorno y a su contexto, derivado de este esfuerzo científico e histórico, se han dado resultados interesantes de estas investigaciones que exponen inherentes relaciones que involucran distintas formas de percepción que parten de un mismo ente. El sonido es un claro ejemplo de este esfuerzo de investigación, entender a este ente natural que integra no solo formas acústicas producidas por las vibraciones de materiales, sino movimientos apreciables en función de esa vibración, esto es lo que condujo en su momento a la fructífera investigación del gran pensador francés Jules Antoine Lissajous, quien en el siglo XIX presentó ante la Academia Francesa los estudios realizados sobre un método para visualizar una curva generada por la combinación de dos movimientos armónicos simples (MAS), en ese momento de la historia, instrumenta este proceso a través de diapasones que logran el reflejo de un haz de luz que permitió dar vida a un concepto abstracto. Para ello, tomó un par de diapasones idénticos, fijó en cada uno un ligero espejo (en uno de los brazos) y reflejó sobre ellos un fino haz de luz perfectamente colimado. Al proyectar este sobre una pantalla, previa reflexión sobre los diapasones en vibración, se formaba una interesante imagen (Boada Ferrer, 2020). Este fue el principio de más instrumentos físicos como péndulos o armonógrafos que permitieron estudiar, crear y recrear a tan maravillosas formas curvas que fueron modeladas en estructuras matemáticas que culminaron en ecuaciones parametrizadas, las cuales permiten hoy día su implementación a través de entornos básicos de programación o simulación matemática.

A partir de lo anterior, pareciera que hablar de procesos matemáticos complejos es algo imposible en el NMS, que pertenecen a un mundo especializado

donde las matemáticas y la informática van hacia un nivel incomprensible, sin embargo, el quehacer del docente corresponde a la simplificación de esta información, no solo como una ocurrencia, sino como un fin que se justifica en la necesidad de enseñar una educación que corresponda con el ODS 4 y la Agenda 2030, donde se asegure la calidad de la enseñanza del uso de la tecnología para todas y todos, con el fin de garantizar que todos los estudiantes posean las capacidades digitales necesarias para tener éxito en el mundo de hoy, impulsado por la tecnología, es de suma importancia (UNESCO, 2023). Por otro lado, es necesaria la correspondencia de fortalecer en los estudiantes los principios básicos de conocimientos científicos y tecnológicos, que atiendan a las necesidades de los nuevos campos de estudio derivados del vertiginoso avance digital en el campo laboral, por ejemplo, para el periodo de 2016-2018, la matrícula en el campo de Ingeniería, Manufactura y Construcción en programas vocacionales disminuyó un 24.5%, mientras que para el campo de Tecnologías de la Información y Comunicación aumentó en más de 1000% y para el campo de Ciencias Naturales, Matemáticas y Estadística aumentó un 83.6% (Baena, 2021).

En correspondencia con el quehacer docente como una respuesta a lo descrito anteriormente, se propone una práctica de laboratorio electrónica-digital (informática) que permita un acercamiento pragmático y controlado para el Nivel Medio Superior (NMS) del Bachillerato Universitario de la Benemérita Universidad Autónoma de Puebla (BUAP), sustentado pedagógicamente en su modelo educativo MUM (Modelo Universitario Minerva), el cual se desenvuelve en un marco constructivista, sociocultural y por competencias, que permite apoyar una propuesta multidisciplinaria que integra electrónica básica, programación, trigonometría elemental y movimiento armónico simple, siendo estos conceptos abordados en el currículo escolar del Bachillerato-BUAP, hecho que permite disponer de los recursos didácticos necesarios para consolidar el proyecto descrito.

OBJETIVOS

Por otro lado, en un sentido por definir las acotaciones pertinentes, así como el espacio de aplicación y desarrollo de propuesta, este trabajo define tres ejes fundamentales, que no significan una delimitación a la creatividad derivada de la propuesta, por el contrario, se establecen las reglas de funcionalidad y sus propósitos pedagógicos, siendo estos los siguientes:

- Desarrollar una práctica de laboratorio (informática) bajo la normativa de una estrategia didáctica de taller.
- La práctica es reproducible, segura y generadora de conocimiento con transversalidad de las áreas que se involucran en su conformación.
- Presentar el marco teórico adecuado al modelo educativo, para su valoración académica en el impacto del perfil de egreso, y su pertinencia con las necesidades científicas y tecnológicas.

DESARROLLO

Presentación de la problemática

Este trabajo responde a la necesidad de atender nuevas realidades educativas, tecnológicas y de emprendimiento, que permitan generar una experiencia de aprendizaje significativo, que aporte al perfil de egreso del estudiante, la formación de un ente competente y responsable en el uso de las tecnologías de la información y comunicación (SEP, 2019). Además, que contribuya al interés por la formación científica, un hecho necesario que hoy en día nos obliga a guiar y a formar jóvenes promesas para la creación de ciencia en nuestro país; un reto que como docentes no es fácil, porque existen limitantes que son propios de los ecosistemas educativos y sus planes de desarrollo; sin embargo, eso no significa que no se puede ir más allá y aportar de manera elemental a un aprendizaje de tipo STEM, que consiste en relacionar la nueva información con las representaciones preexistentes, propiciando aprendizajes significativos en las asignaturas involucradas en la consolidación de esta

práctica.

Por otro lado, la propuesta también atiende al llamado de la generación del conocimiento y del quehacer docente, ante una situación que observa una brecha del 54.3% entre el número de solicitudes y las patentes efectivamente otorgadas. En cuanto a otro tipo de resultado científico, para el periodo de 2014-2018, la producción de artículos científicos llevados a cabo por investigadores mexicanos con relación al resto de los países miembros de la OECD se ubicó en el lugar 20 de las 36 naciones integrantes (Baena, 2021).

FUNDAMENTACIÓN TEÓRICA

Las necesidades de conocimiento sobre nuevas tecnologías, la disposición de un modelo educativo enfocado a la formación STEM bajo las directrices del modelo educativo MUM-BUAP y por Competencias, permite el desarrollo de los conceptos y campos de acción que guían a la transversalidad entre las áreas de Matemáticas e Informática, que a su vez correspondan al tipo de formación e información que es utilizada para consolidar esta propuesta a partir de los principios básicos, considerando que los conceptos expuestos no rebasan al entorno de desarrollo del NMS, evitando caer en tecnicismos innecesarios que pudieran confundir o complicar tanto al docente como al estudiante por su grado de dificultad, en el ejercicio de este proyecto electrónico-digital.

METODOLOGÍA

El presente trabajo se desarrolla bajo la implementación de una estrategia didáctica de tipo taller, el cual permite un aprendizaje integral del conocimiento ya que “se trata de una forma de enseñar y, sobre todo de aprender, mediante la realización de algo que se lleva a cabo conjuntamente” (Anger-Egg, 1999). La secuencia de pasos y su interacción en la construcción del conocimiento, “es un aprender haciendo, en el que los conocimientos se adquieren a través de una práctica sobre un aspecto de la realidad” (Anger-Egg, 1999), también es importante

recalcar que este trabajo persigue acercar al estudiante a la necesidad del aprendizaje de la codificación, es decir, de la programación de computadoras, ya que “en el proceso de aprendizaje de la codificación, la gente aprende muchas otras cosas. No solo aprenden a codificar; codifican para aprender” (Resnick, 2013), un hecho que aporta al bagaje cultural y formativo de los estudiantes, ante un mundo digital cambiante y obligado por las circunstancias actuales, donde los ordenadores se han convertido en escuela, trabajo, medio y desde luego, generadores de ciencia.

En el campo matemático se propone manejar un plano cartesiano, que hace uso de la ecuación parametrizada de las curvas de Lissajous, esta interpretación matemática permite un cómodo manejo de los datos y facilita la movilidad de saberes previos en los estudiantes, por su manejo elemental de dos funciones trigonométricas -seno y coseno-. La decisión de utilizar esta ecuación atiende a su importancia en la vida diaria de la formación académica y a su existencia en la realidad misma, ya que es objeto de estudio en la física clásica, lo electrónico y las matemáticas.

En esta estrategia didáctica se establece una secuencia simplificada de pasos que lleva por propósito ensamblar un dispositivo electrónico básico a partir de una interfaz Arduino y dos palancas de juego tipo joystick, que permitirán una transmisión de datos a una conexión en el puerto serial (COM#), que recibirá datos que serán procesados mediante una librería especializada para el software matemático (Octave) que funciona como enlace entre el mundo exterior y el control digital, estos datos serán enviados a la ecuación parametrizada de Lissajous (1), en

específico a los valores w_1 y w_2 de esta, quienes generan peculiares comportamientos en la generación de las curvas; pero para ello, es necesario entender de manera general cómo funciona esta parametrización, su comportamiento son dos movimientos armónicos simples con ecuaciones desacopladas (Álvarez Ramírez & García, 2013) por un desfase (δ), que permiten la producción de curvas cerradas cuando el movimiento es periódico a partir de un valor racional generado por w_1/w_2 .

$$\text{Sea } \begin{cases} x(t)=\text{sen}(w_1 t) \\ y(t)=\text{sen}(w_2 t+\delta) \end{cases} \quad (1)$$

Donde su dominio se establece en un intervalo $0 \leq t \leq 2\pi$, con δ como diferencia de fase delimitada en una acotación $0 \leq \delta \leq \frac{\pi}{2w_1}$.

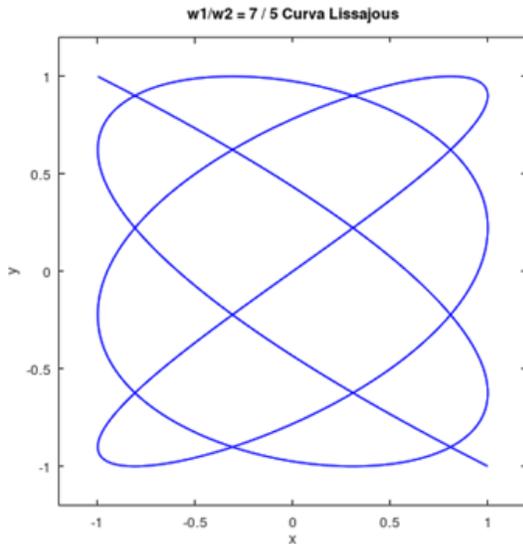
En el marco teórico matemático, se asiste de conocimientos elementales para un mayor aprendizaje por parte de los estudiantes, a través de la exposición de ejemplos de caso con funciones parametrizadas que atienden al comportamiento de los cambios de las variables, y que proporcionan un acercamiento a formas más allá de las tradicionales descripciones de las funciones seno y coseno. Las curvas de Lissajous, presentan un comportamiento de trazo determinados por los valores de los parámetros w_1 y w_2 , que a su vez dependen también de la distancia de desfase δ .

En los ejemplos expuestos en la tabla 1, se aprecia una curva no cerrada con valores $w_1=7$ y $w_2=5$ y un desfase δ de 0 (figura 2), ahora, intercambiando estos valores w_1 y w_2 , donde se aprecia la diferencia en sus formas trazadas (figura 3).

Tabla 1. Curvas de Lissajous con desfase δ de 0.

$w_1 = 7, w_2 = 5 \rightarrow 7/5$

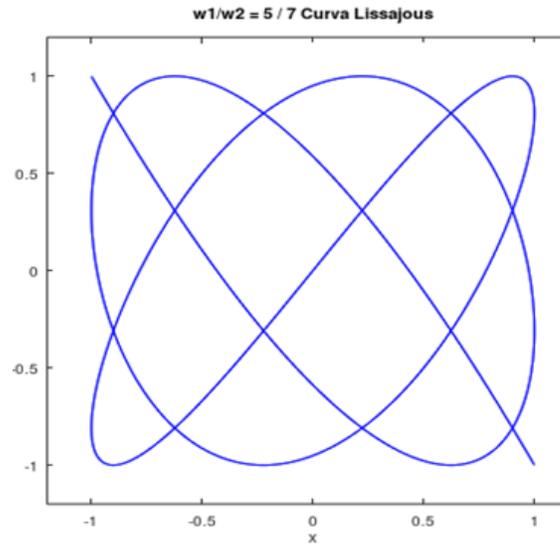
Figura 2. Curva Lissajous (7/5)



Fuente: Elaborado en Octave

$w_1 = 5, w_2 = 7 \rightarrow 5/7$

Figura 3. Curva Lissajous (5/7)



Fuente: Elaborado en Octave

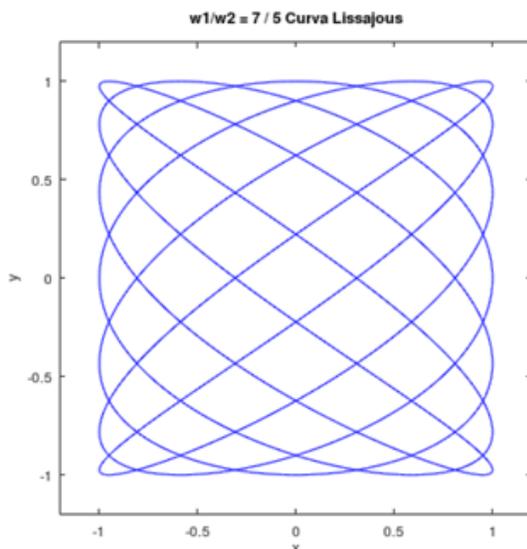
En el siguiente ejemplo de la tabla 2, se presenta la figura 1, se aprecia una curva no cerrada con valor

$w_1 = 7$ y $w_2 = 5$ (figura 4), con un desfase δ de $\pi/2$, y otro en intercambio de valores $w_1 = 5$ y $w_2 = 7$ (figura 5).

Tabla 2. Curvas de Lissajous con desfase δ de $\pi/2$

$w_1 = 7, w_2 = 5 \rightarrow 7/5$

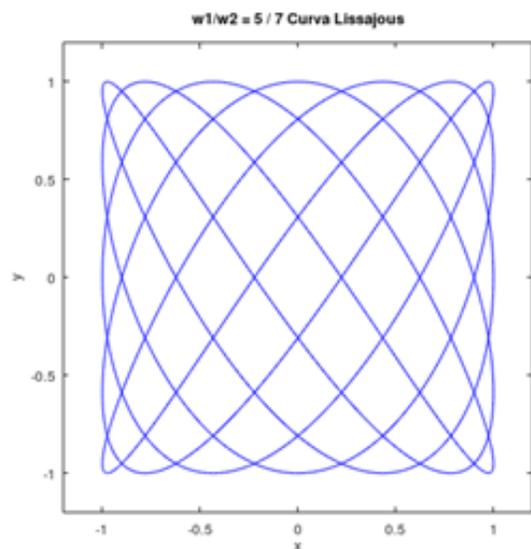
Figura 4. Curva con desfase δ de $\pi/2$



Fuente: Trazado en Octave

$w_1 = 5, w_2 = 7 \rightarrow 5/7$

Figura 5. Curva con desfase δ de $\pi/2$

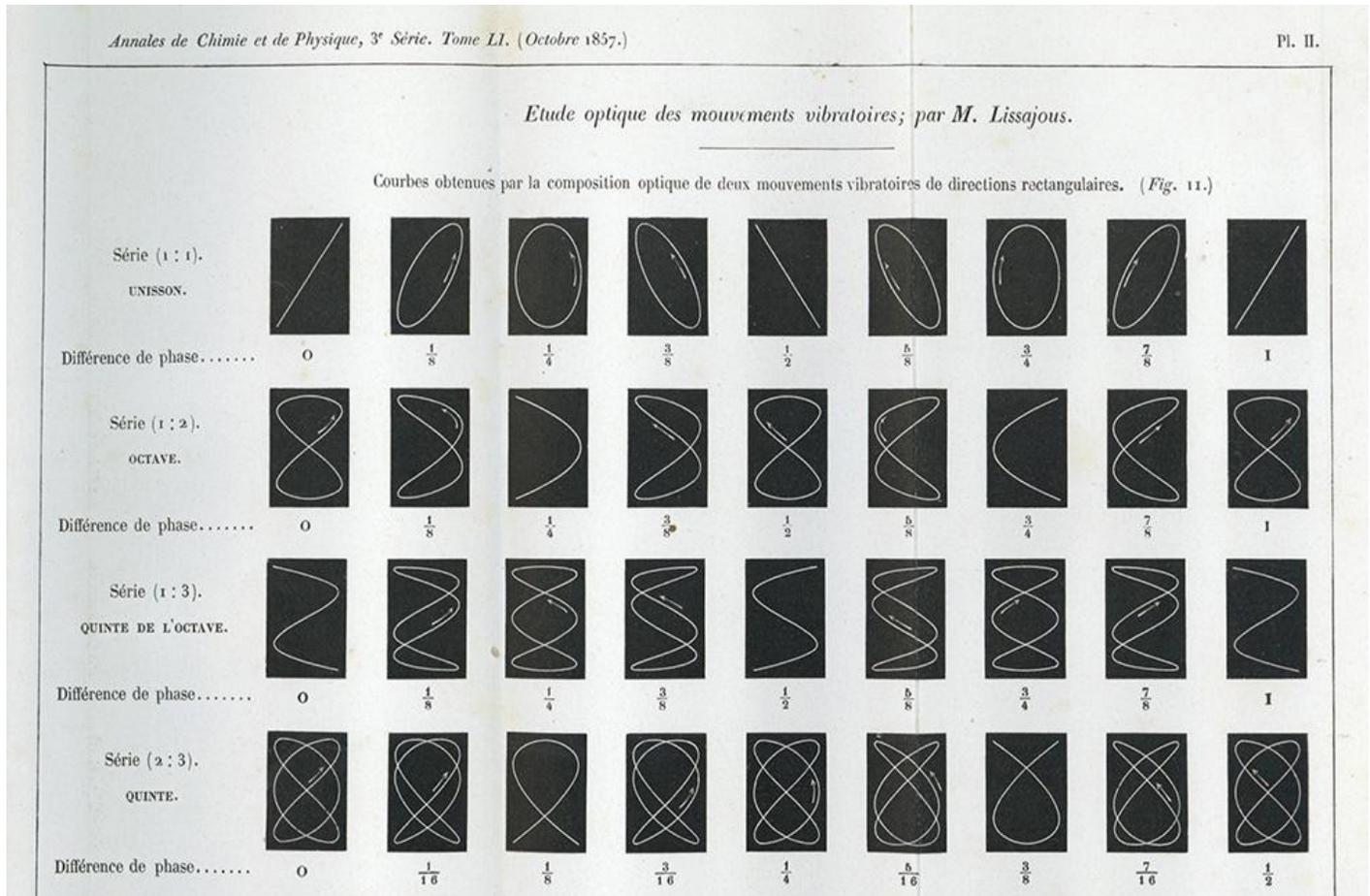


Fuente: Trazado en Octave

En la actualidad nos sorprendemos de los bellos trazos generados por las ecuaciones y su disponibilidad en programas informáticos. No obstante, lo que más sorprende es que en octubre de 1857, sin software ni medio digital

sofisticado, Lissajous presentó una tabla (véase figura 5) con las formas observadas en el apartado denominado "Etude optique des mouvement vibratoires" que corresponde al texto "Annales de chimie et de physique" (physique, 2020).

Figura 5. Trazos de Movimientos vibratorios obtenidos por Lissajous.

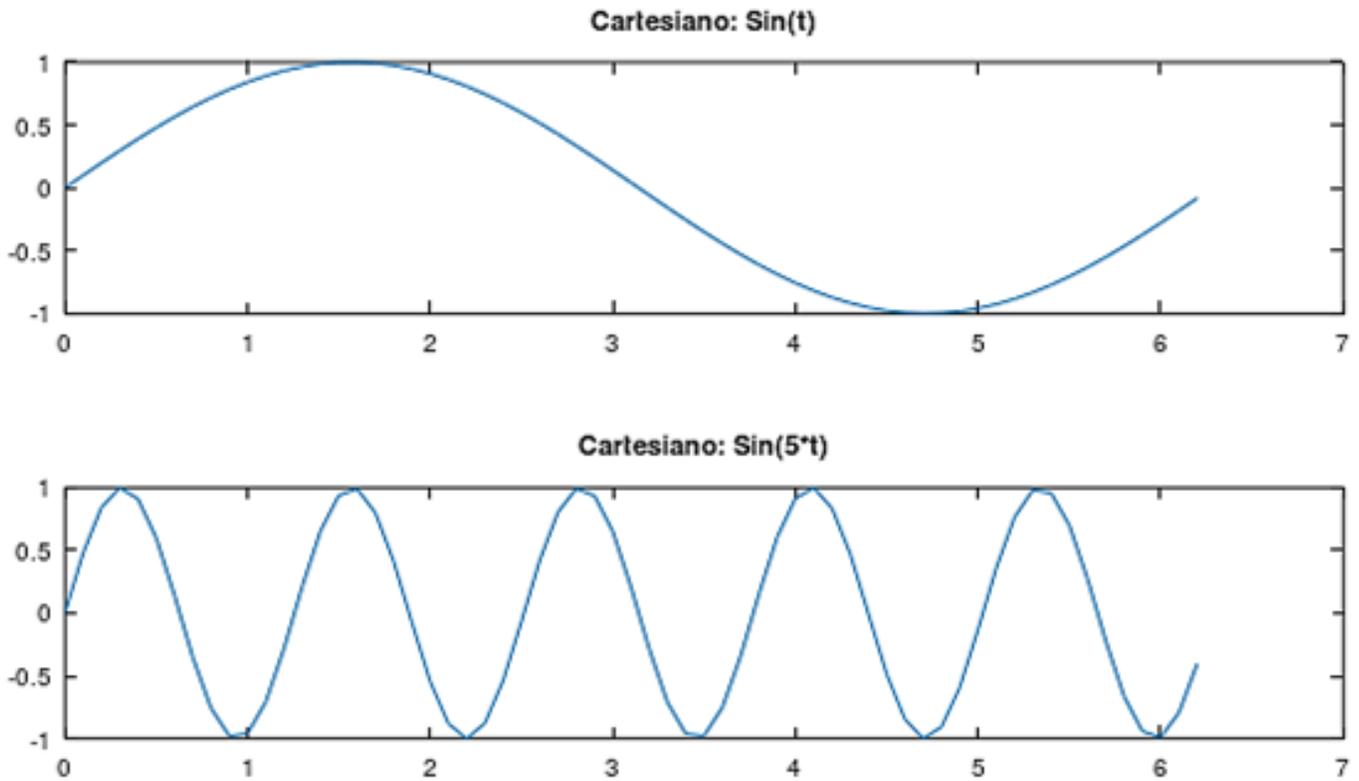


Fuente: <https://gallica.bnf.fr/ark:/12148/bpt6k65708374/f519.item>

Paso 1. Interpretación matemática y su implementación en el software.

El primer esfuerzo para desarrollar esta propuesta es exponer ante el estudiante, la interpretación matemática

de las curvas de Lissajous, la importancia del desfase de las curvas y la peculiaridad de los formas racionales e irracionales que se dan al considerar los parámetros de amplitud de las funciones trigonométricas (figura 6).

Figura 6. Amplitud de las funciones trigonométricas.

Fuente: Trazado propio en el Software Octave

A partir de la parametrización cartesiana (2) de las curvas de Lissajous y de identificar su comportamiento en cuanto a los valores racionales e irracionales generados por los valores w_1 y w_2 , así como al desfase δ .

$$\text{Sea } \begin{cases} x(t)=\text{sen}(w_1 t) \\ y(t)=\text{sen}(w_2 t+\delta) & 0 \leq \delta \leq \pi/2w_1, (2) \\ 0 \leq t \leq 2\pi \end{cases}$$

El software de desarrollo corresponde a un programa de código abierto, que sigue estándares de programación semejantes a lenguajes de importante renombre como Matlab, de hecho, Octave tiene una gran similitud con el manejo de sintaxis y estilo de programación, lo cual facilitará al estudiante una migración factible al sistema comercial, lo que reduce la curva de aprendizaje y

adaptación para el aprendizaje de programación básica.

La primera etapa consiste en identificar el comportamiento de la curva de Lissajous al modificar el desfase, con valores como $\delta=0$ y $\delta=\pi/2$, lo que produce dos trazos distintivos, el primero de ellos una curva abierta (véase figura 6) y otra cerrada (véase figura 6), por otro lado, existe una aportación importante que considerar si el trazo se realiza bajo la ecuación 2, y es el hecho de que la forma $\pi/2w_1$, sigue el mismo patrón de trazado.

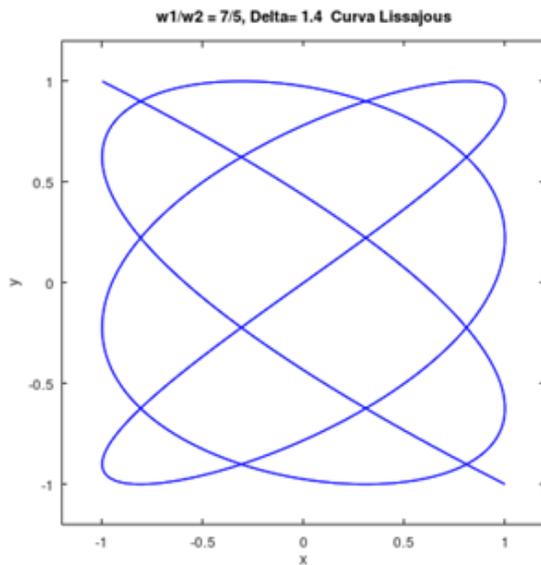
A continuación, se presentan los gráficos generados a partir de lo descrito anteriormente, donde se puede apreciar la comparativa entre las ecuaciones 1 y 2. Esto permite observar la importancia de los parámetros w_1 y w_2 , como referentes fraccionarios w_1/w_2 ($1 \leq w_1 \leq w_2$), a partir de su clasificación racional o irracional (Ferréol, 2017).

Cociente racional

Sean $w_1 = 7$ y $w_2 = 5$, con desfase $\delta = 0$ (figura 7) y $\delta = \pi/2$ (figura 8), la generación de dos trazos distintos bajo la

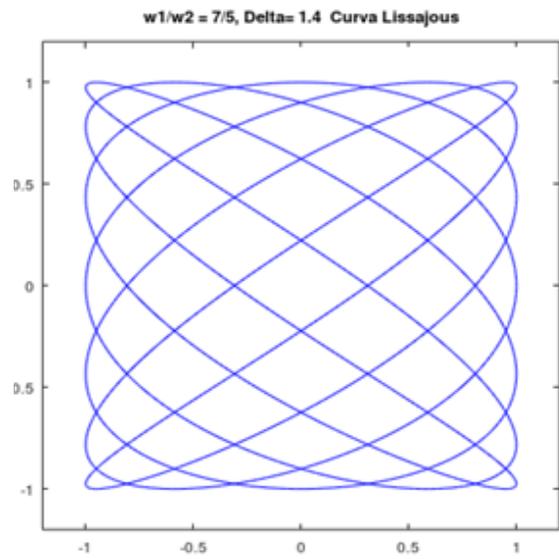
ecuación 1, también responde con la misma forma de la ecuación 2, con cociente racional 7/5 (1.4), desarrolla una curva abierta en el primer caso y cerrada en el segundo.

Figura 7. 7/5 con desfase $\delta = 0$



Fuente: Trazado en Octave

Figura 8. 7/5 con desfase $\delta = \pi/2$



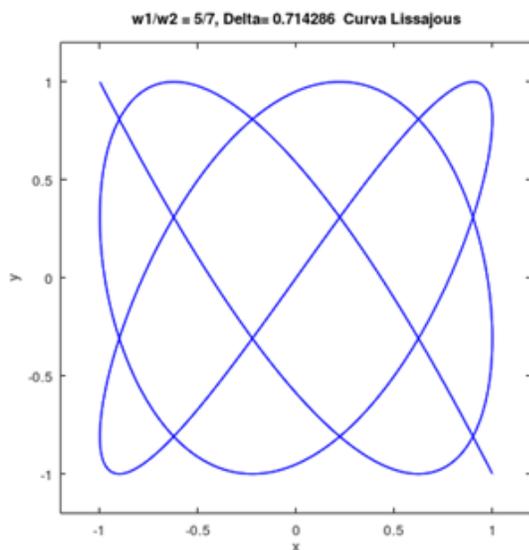
Fuente: Trazado en Octave

Finalmente toca turno a considerar un intercambio de parámetros w_1 y w_2 , que reafirman la importancia del cociente.

(figura 10), con cociente irracional 5/7=; se aprecia una curva abierta para el primer desfase y una cerrada en el caso del segundo valor de delta.

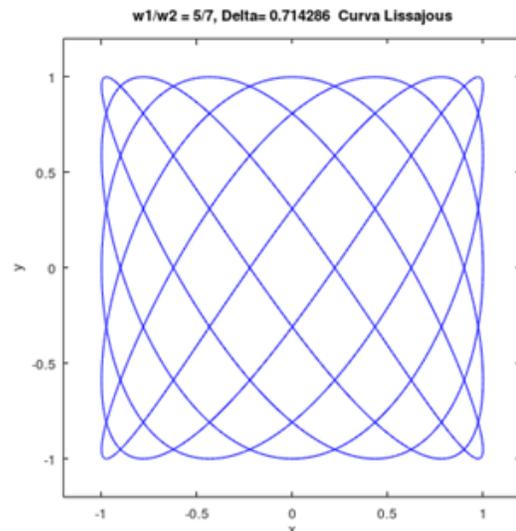
Sean $w_1 = 5$ y $w_2 = 7$, con desfase $\delta = 0$ (figura 9) y $\delta = \pi/2$

Figura 9. 5/7 con desfase $\delta = 0$



Fuente: Trazado en Octave

Figura 7. 5/7 con desfase $\delta = \pi/2$



Fuente: Trazado en Octave

Para la codificación es necesario hacer uso del software denominado Octave, desarrollado en 1988, por John W. Eaton y colaboradores, el cual tiene una versión estable verificada en febrero de 2020. Este software es considerado el equivalente a Matlab (software comercial de enfoque matemático) en su contexto GNU (W. Eaton, 2020), ya que contiene estructuras sintácticas semejantes y compatibles con las instrucciones. Gracias a esto, se tiene un software que no implica gasto en licencias y el aprendizaje de su programación aportarán también conocimientos para Matlab, disminuyendo la curva de aprendizaje de este, y su migración en futuros proyectos profesionales.

En la tabla 1, se presenta el código fuente para la generación de las curvas de Lissajous y sus parámetros, la facilidad de intercambio en los datos se observa al definir las variables w_1 y w_2 para la amplitud y el desfase definido por la variable denominada delta. El rango de trazo esta definido por la instrucción *linspace()*, que parte de 0 a 2π , utilizando 1000 puntos para la generación del gráfico. El movimiento está definido por las variables x e y, que se ajustan a las ecuaciones de la forma cartesiana; por último, el trazo y ajuste de los ejes a través de las instrucciones *plot* y *axis* respectivamente. Las instrucciones *xlabel* y *ylabel* integran cadenas ilustrativas para la gráfica, y *str* contiene una cadena que será incorporada como título.

Tabla 1. Código fuente para la generación de curvas de Lissajous

<pre>w1 = 7; _____ % Valor w1 w2 = 5; _____ % Valor w2 %1000 puntos para un rango de 0 a 2pi (dominio) t = linspace(0,2*pi,1000); delta=pi/2; _____ % intercambio de desfases 0 / 2Pi x = sin(w1*t); _____ % x Movimiento y = sin(w2*t+delta); _____ % y Movimiento plot(x,y,'b-') _____ % plot Curva color-azul axis equal _____ % Ejes escalados al mismo tamaño axis([-1.2 1.2 -1.2 1.2]); % Ajuste de rango en el ploteo % Etiquetas y título xlabel('x'); ylabel('y'); str = sprintf('w1/w2 = %i / %i Curva Lissajous',w1,w2); title(str);</pre>	<p>Parámetros de Variación</p> <p>w_1</p> <p>w_2</p> <p>delta (desfase)</p>
---	---

Como se ha podido apreciar en la codificación mostrada en la tabla anterior, desarrollar el modelo cartesiano en Octave

no implicó gran especialización, se utilizaron instrucciones básicas que se asocian con aprendizajes básicos.

Paso 2. La interfaz electrónica

La siguiente etapa consiste en desarrollar la interfaz electrónica digital que genera datos que son incorporados como valores de los parámetros w_1 , w_2 , y el desfase delta (δ), a través de la lectura del puerto serial COM#. Para desarrollar este circuito se hace uso de la placa Arduino Uno, un dispositivo electrónico-digital de licencia libre (GNU) creado en 2005, como un proyecto nacido de una tesis para la construcción de una placa de trabajo académico con un menor, para apoyar a los estudiantes (Díaz Díaz, 2014) , desde esta aventura, hasta hoy en día, este dispositivo está revolucionando el desarrollo de prototipos que han consolidado a importantes prototipos de hardware tanto en la empresa como en la educación.

La programación previa a la integración directa al microprocesador, tiene instrucciones y sintaxis

similar a la de lenguaje C, así que su aprendizaje no significará mayor problema al momento de su explicación. Las instrucciones que se utilizan se muestran en la tabla 2, donde se especifica la utilidad de cada una de las funciones para la programación del microprocesador.

A la tarjeta Arduino, se le incorpora un dispositivo de interfaz humana, que en este caso corresponde a dos palancas de juego (joystick), las cuales se conectan a las entradas analógicas x_1 y x_2 , para el ingreso de los valores x_1 de la primera palanca asignada al parámetro w_1 , y la segunda palanca definida como x_2 para w_2 . Estos valores son integrados a la placa y filtrados a través de la salida serial con formato de cadena, véase en la tabla 3, la codificación completa.

Tabla 2. Funciones utilizadas en la interfaz electrónica-digital

<i>Función</i>	<i>Utilidad</i>
<code>pinMode(PIN, INPUT);</code>	<i>Modo de Pin (Entrada-Input)</i>
<code>digitalWrite(PIN, HIGH);</code>	<i>Escritura directa sobre un Pin (HIGH-Voltaje)</i>
<code>Serial.begin(9600);</code>	<i>Velocidad en baudios del puerto serial (COM#)</i>
<code>analogRead(PIN)</code>	<i>Lectura del Pin analógico</i>
<code>Serial.print("cadena", valor)</code>	<i>Escritura serial directa como cadena</i>

Tabla 3. Codificación de la interfaz de Arduino

```

// Código para interfaz (Palancas joysticks)

//Pin Digital 2: Palanca Num=>Presionar (DESFASE)
const int Press_pin = 2;

//Pin Analógico 1: Palanca Num1=>X1=>W1
const int X1_pin = 1;

//Pin Analógico 2: Palanca Num2=>X2=>W2
const int X2_pin = 2;

void setup() {
//Modo entrada digital
pinMode(Press_pin, INPUT);
//Valor 1 para desfase =0 / Valor 0 para desfase =pi/2
digitalWrite(Press_pin, HIGH);
Serial.begin(9600); // Baudios (Vel.)
}
void loop() {
// 1023 valores en palanca / 128 =>
// Lectura e impresión de datos de entradas análogas
// Salida generada a través de la función print
// Generación de cada palanca con valores X1 y X2
// Integración a la cadena del evento de presión.
Serial.print(analogRead(X1_pin)/64);
Serial.print(",");
Serial.print(analogRead(X2_pin)/64);
Serial.print(",");
Serial.print(digitalRead(Press_pin));
Serial.print("\n");
}

```

Los valores son capturados a través de la lectura de los valores eléctricos generados por las palancas; estas señales son registradas en las entradas analógicas, presentando un rango de 1024 valores (0 a 1023).

Los valores registrados son divididos en 64 para acotar el valor de los parámetros $w1$ y $w2$. Los resultados son canalizados como una cadena en la señal serial del puerto COM.

Paso 3. Diseño del prototipo en la placa Arduino

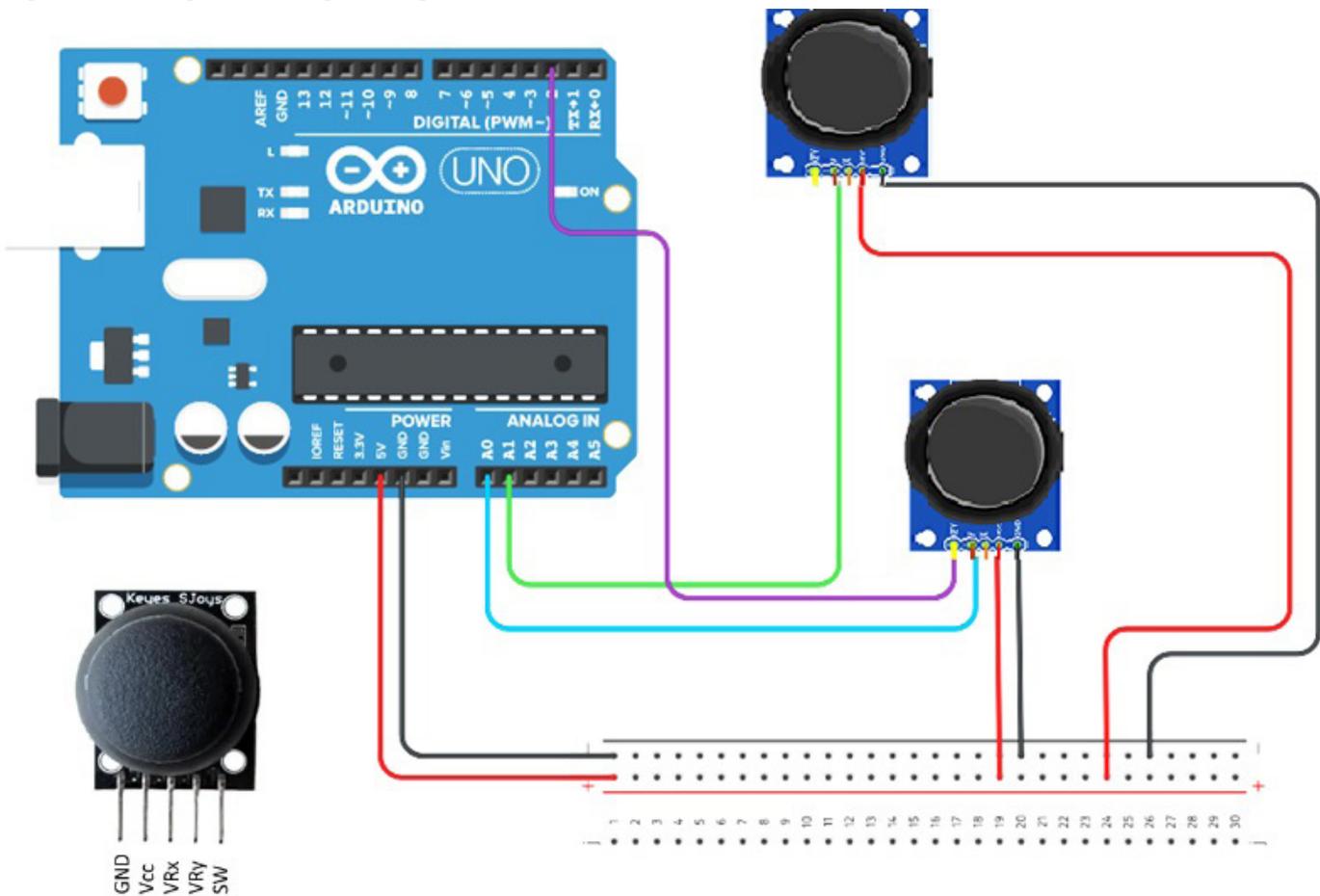
El desarrollo del dispositivo electrónico integra de manera simplificada la incorporación de dos palancas joystick, en las entradas analógicas A1 ($w1$), A2 ($w2$) y el cambio de desfase en la entrada

digital D2, esta última para controlar el cambio de trazo bajo un desfase $\delta=0$ y $\delta=\pi/2$. Posteriormente, se conecta a puerto COM#, se procede a la carga del script en el procesador y este genera datos a través del ciclo (loop) de la instrucción dada en el código (Tabla 4).

Tabla 4. Materiales para la interfaz Arduino	
Una placa Arduino UNO	
Dos palancas tipo joystick	
Un protoboard, cableado Dupont	

Para el armado de la interfaz, se presenta la figura 11, la cual presenta el cableado y orden que deberá seguir cada palanca y su transmisión de datos a través de los pines analógicos.

Figura 11. Esquema de prototipo



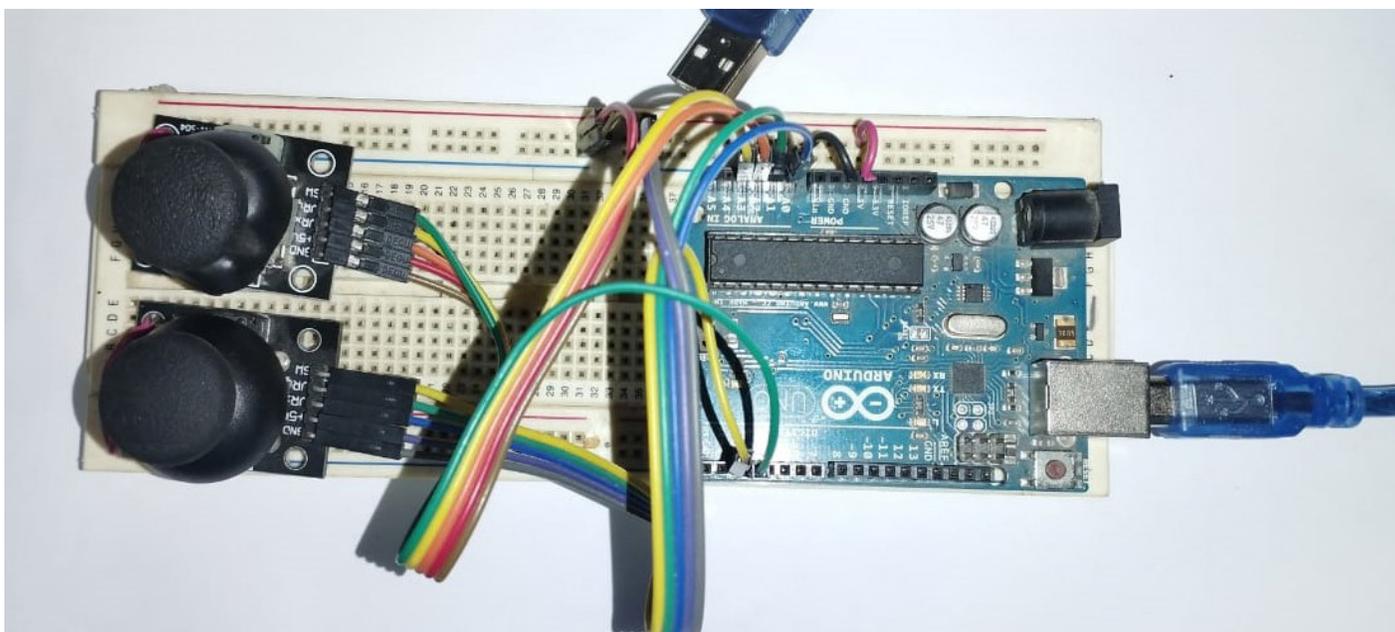
Fuente: Diseño propio en herramienta básica

Los datos se integran a la placa Arduino mediante el movimiento del Joystick, este proporciona los valores de las coordenadas X e Y, además del valor indicativo de la palanca presionada. Estos valores son de tipo analógico, tienen un rango de 0 a 1023 y para esta propuesta los valores han sido ajustados a valores más bajos en la estructura de programación, de esta manera el movimiento de la palanca se canalizan en Arduino a través de la comunicación serial, los cuales son recibidos por la interfaz COM#, donde Octave

a través de las instrucciones para el manejo de entradas como cadenas de datos, realiza un proceso de separación, donde se ajusta en arreglos las variables de posición X e Y que funcionan como argumentos para la ecuación de Lissajous (2), el diseño final se muestra en la figura 12, el cual está compuesto por:

1. Placa Arduino UNO R3.
2. Protoboard, cableado.
3. Joystick.

Figura 12. Prototipo físico de la interfaz digital Arduino-Joystick



Fuente: Elaboración propia del prototipo electrónico-digital

Paso 4

El software de interfaz y el manejo de los datos

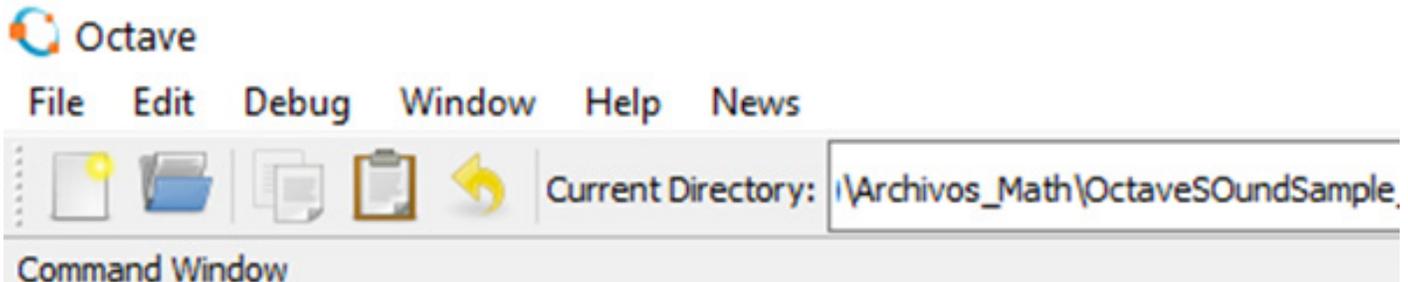
En cuanto al software de apoyo para realizar el trazado de las curvas de Lissajous, se opta por Octave, una versión libre semejante a Matlab, esta semejanza permite una rápida adecuación para quienes ya conocen este lenguaje; y quienes inician tendrá la oportunidad de aprender un estándar de las matemáticas.

Para realizar la integración de la interfaz de Octave con la plataforma física de Arduino, se presentaron dos opciones, en la primera se pudo observar cierta facilidad para procesar de manera directa la lectura

de datos a través de la carga de un script supervisor que permite la integración directa en Octave, para ese proceso fue necesario la instalación de librerías especializadas a través de otros “proveedores” de apoyo en la comunidad, en este caso la programación fue realizada por John Donoghue (Donoghue, s.f.), quien presenta el script y toda la documentación necesaria con licencia GPLv3+ a través del sitio Octave Forge.

La instalación del paquete para este caso fue vía online, una vez descargado el software de su sitio oficial (<https://www.gnu.org/software/octave/index>) y ejecutado, se procede con las siguientes instrucciones:

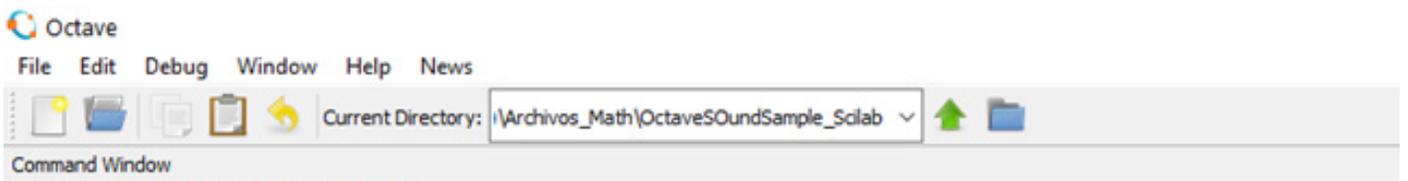
1. En consola (véase figura 13) escribir la instrucción `pkg install -forge arduino`, directamente sobre la ventana de comando.



```
>> pkg install -forge arduino
```

Fuente: Entorno Octave

2. Para la carga del paquete en cada sesión de GNU Octave, escribir la instrucción: `pkg load Arduino` (véase figura 14).



```
>> pkg install -forge arduino
For information about changes from previous versions of the arduino package, run 'news arduino'.
>> pkg load arduino
>> |
```

Fuente: Entorno Octave

3. El siguiente paso es programar en Arduino, para ello se puede ejecutar la instrucción `arduinsetup`, la cual abre la plataforma Arduino y presenta un script que al cargarlo directamente sobre la placa Arduino, se dispone de una conexión directa vía consola para programar, sin embargo, para este caso, se ha dispuesto de una programación apoyada en el editor de Octave.

Con los anteriores pasos se tiene una codificación más directa entre octave y Arduino, sin embargo, en este trabajo, no fue posible amoldar las instrucciones necesarias para generar las curvas de Lissajous, ya que las lecturas de los datos generados por las palancas integraban información inadecuada en el trazado, produciendo líneas rectas que no corresponden a los resultados de dos funciones trigonométricas.

A partir de la experiencia anterior, se opta por un modelo de programación que permita leer los datos directamente desde el puerto COM# como un resultado de cadena serial, lo que facilita independizar la programación de la interfaz Arduino y los datos que se integran a Octave. Para realizar el proceso descrito anteriormente, se hace uso de una librería denominada `instrument-control` (<https://octave.sourceforge.io/instrument-control/overview.html>), la cual contiene instrucciones de bajo nivel para acceso a interfaces seriales, i2c, paralelas, tcp y otras, con ello se puede leer directamente sobre el buffer del puerto COM# y extraer la información como una cadena que se “desencripta” para separar los datos transportados por Arduino.

Procesamientos de datos

En general se realiza la programación a partir de la lectura directa del buffer del puerto COM#, a través de las instrucciones apropiadas que se

muestran en la tabla 5, donde se denotan las fases de conexión, lectura, conversión, integración al modelo matemático, trazado gráfico y cierre del canal de datos COM#.

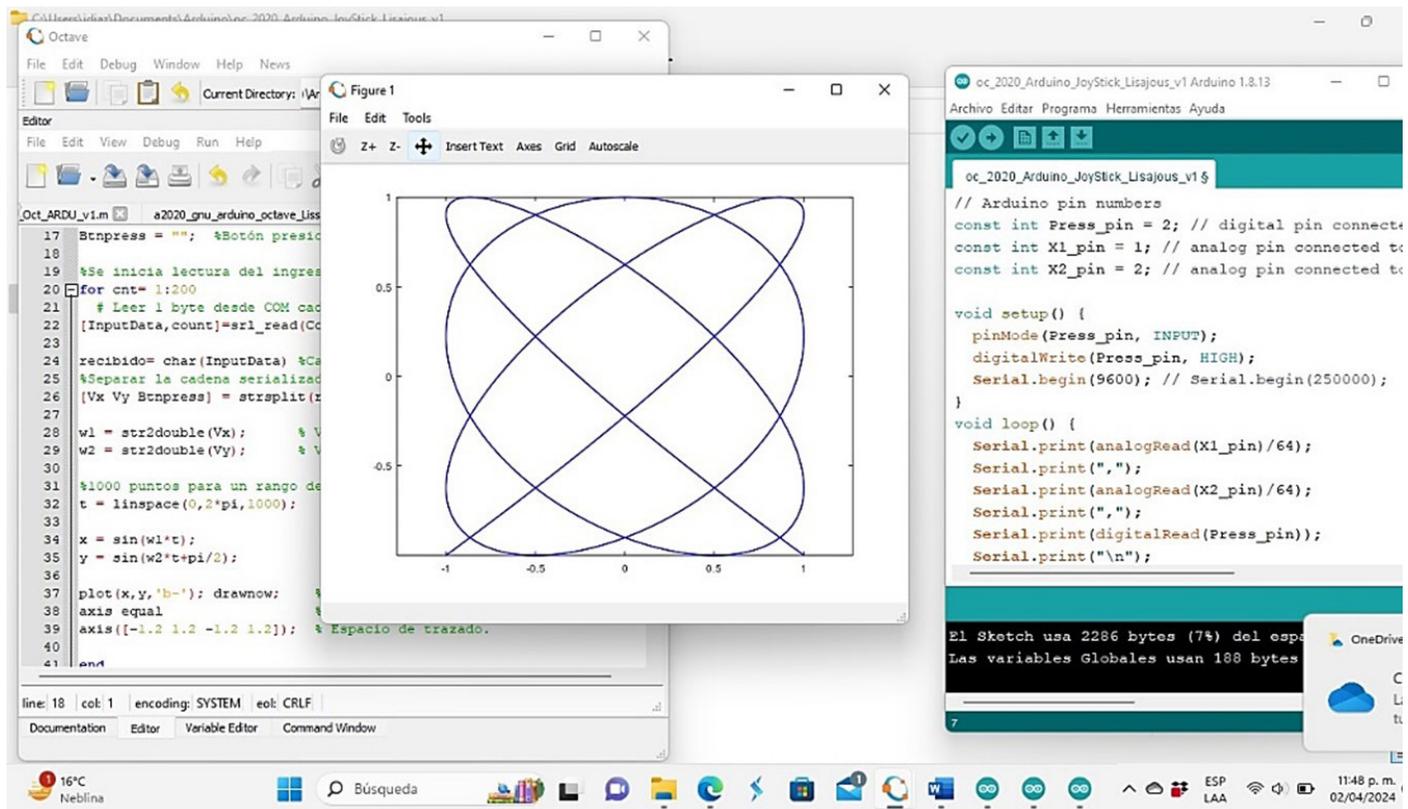
Tabla 5. Codificación de Octave y su interfaz con Arduino	
<pre>clc;clear; pkg load instrument-control # Carga control de instrumentos ConectPort=serial("COM3"); #Puerto COM3 #Pausa de 1 segundo pause(1); # Configuración SERIAL set(ConectPort,'baudrate',9600); set(ConectPort,'bytesize',8); %Tamaño 1 Byte set(ConectPort,'parity','n'); set(ConectPort,'stopbits',1); set(ConectPort,'timeout',50); % Tiempo fuera srl_flush(ConectPort); % Inicia lectura de flujo Vx = ""; %Valor Palanca A (Vx) como cadena Vy = ""; %Valor Palanca B (Vy) como cadena Btnpress = ""; %Botón presionado Palanca A %Se inicia lectura del ingreso de datos por el canal COM# for cnt= 1:100 # Leer 1 byte desde COM cada 200ms</pre>	<p>Conexión con el puerto COM#</p>
<pre>[InputData,count]=srl_read(ConectPort,200); recibido= char(InputData) %Caracter recibido %Separar la cadena serializada en tres variables [Vx Vy Btnpress] = strsplit(recibido, ","){1,:}</pre>	<p>Configuración de lectura del puerto COM#</p>
<pre>w1 = str2double(Vx); % Valor w1 (convertir a número double) w2 = str2double(Vy); % Valor w2 (convertir a número double)</pre>	<p>Asignación de información serial (cadena) Lectura de la cadena.</p>
<pre>%1000 puntos para un rango de 0 a 2pi (dominio) t = linspace(0,2*pi,1000); x = sin(w1*t); % Coordena X y = sin(w2*t+pi/2); % Coordena Y</pre>	<p>Separación de la cadena para su procesamiento como números</p>
<pre>plot(x,y,'b-'); drawnow; % plot Curva color-azul / trazo inmediato axis equal % Escalado igual axis([-1.2 1.2 -1.2 1.2]); % Espacio de trazado. end</pre>	<p>Conversión de información serial (cadena)</p>
<pre>fclose(ConectPort); %Cierra el Puerto COM3</pre>	<p>Integración al Modelo matemático.</p>
	<p>Trazado de las curvas de Lissajous</p>
	<p>Cierre de lectura del puerto COM#</p>

DESARROLLO

El desarrollo de la propuesta exhibe los pasos necesarios para realizar una práctica reproducible (véase fig. 15), con un claro seguimiento y resultado, los materiales electrónicos no presentan un riesgo físico, dado la calidad de la tarjeta Arduino UNO R3 y el montaje del prototipo. Con lo anterior se tiene un proyecto integrador que puede ser implementado y evaluado dentro del Nivel Medio Superior, a partir del

sexto semestre, donde los conocimientos de informática y matemáticas pueden coincidir, propiciando en el estudiante una experiencia de aprendizaje significativo. Apoyarse en este trabajo, también reducirá la búsqueda de información y pruebas de campo, ya que no se cuenta con suficiente información en el caso de interfaces electrónicas-digitales con Octave y la placa Arduino, dado que es un campo de reciente aplicación, pero que ha avanzado a pasos agigantados por la necesidad de integrar datos generados en tiempo real a modelos matemáticos.

Figura. 15. Curva de Lissajous generadas por la propuesta utilizando dos palancas JoyStick.

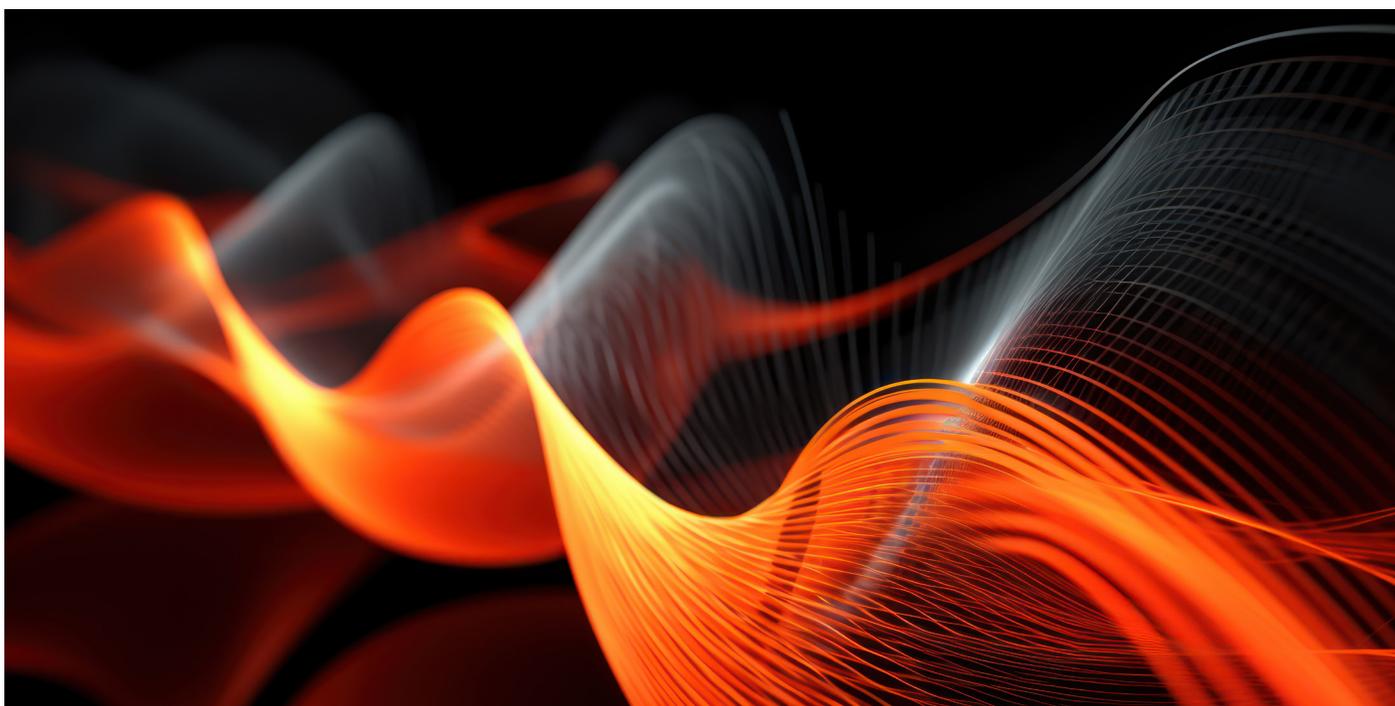


Fuente. Diseño procedente de la propuesta.

DISCUSIÓN

En un marco de desarrollo previo y de pruebas de la propuesta presentada en este trabajo, se observa que la ejecución de la aplicación responde adecuadamente a su propósito, la generación de curvas de Lissajous (véase fig. 15) utilizando dos palancas para modificar los parámetros w_1 y w_2 de la ecuación (2), presentan

una interacción entre un modelo matemático y las variables externas generadas a partir de datos digitales que controla el usuario, presentando formas apreciables en la interfaz gráfica que se coordina con Octave, este significa que la secuencia de desarrollo de la práctica de laboratorio es reproducible, asimismo los estudiantes con los conocimientos elementales de trigonometría pueden incorporar este aprendizaje en el sexto semestre del Bachillerato General-BUAP .



CONCLUSIÓN

Al realizar este proyecto se encontraron algunas problemáticas en la programación, la interfaz y documentación para su desarrollo, dado la escasa existencia de información sobre la conexión de interfaces electrónicas con Octave, esto a pesar de que existen librerías y “plugins” desarrollados por toda una comunidad, así como el soporte que da la organización. Poco se ha desarrollado en esta área con respecto a empresas como Matlab, que tienen equipos profesionales de trabajo

desarrollando estas interfaces desde hace muchos años. Sin embargo, con la información que se tiene es posible llegar a consolidar proyectos como el presente trabajo, donde se pretende enseñar al estudiante la importancia del manejo de datos generados a partir de ambientes externos en tiempo real y sobre todo, es una oportunidad para aprender a construir interfaces digitales básicas que permitan integrar datos masivos. En Internet es un poco difícil encontrar aportaciones que expongan un manejo como el trabajo presentado, por lo que esta propuesta es una contribución a la comunidad que maneja este software y en el especial a las licencias libres.

BIBLIOGRAFÍA

- Álvarez Ramírez, M., & García, A. (2013).** El movimiento oscilatorio. (D. d. Matemáticas, Ed.) Contactos, Revista de Educación en Ciencias e Ingeniería(90), 48. Recuperado el 15 de marzo de 2020, de <http://www2.izt.uam.mx/newpage/contactos/revista/90/pdfs/oscilatorio.pdf>
- Anger-Egg, E. (1999).** El taller una alternativa de renovación pedagógica. Buenos Aires: Magisterio del Río de la Plata.
- Baena, G. (2021),** Indicadores STEM para México, Movimiento STEM, Ciudad de México.
- Boada Ferrer, M. (2020).** Las figuras de Lissajous. Investigación y Ciencia(553), 1. Recuperado el 20 de mayo de 2020, de <https://www.investigacionyciencia.es/revistas/investigacion-y-ciencia/el-nuevo-coronavirus-796/las-figuras-de-lissajous-18475>
- Díaz Díaz, J. (2014).** Cinco Días. (E. País, Editor) Recuperado el 15 de enero de 2020, de https://cincodias.elpais.com/cincodias/2014/03/31/mwc/1396267341_826466.html
- Donoghue, J. (s.f.).** Octave Arduino Toolkit. (O. Forge, Editor) Recuperado el 10 de 12 de 2019, de <https://octave.sourceforge.io/arduino/index.html>
- Ferréol, R. (2017).** mathcurve.com. Recuperado el 14 de 1 de 2020, de <https://mathcurve.com/courbes2d.gb/lissajous/lissajous.shtml>
- Physique, A. d. (12 de marzo de 2020).** Gallica. Obtenido de <https://gallica.bnf.fr/ark:/12148/bpt6k34792z/f513.image>
- Resnick, M. (2013).** Learn to Code, Code to Learn. EdSurge.
- UNESCO. (2023).** Educación de niñas y mujeres en ciencia, tecnología, ingeniería y matemáticas (STEM). <https://www.unesco.org/es/gender-equality/education/stem>
- W. Eaton, J. (2020).** GNU Octave. Obtenido de <https://www.gnu.org/software/octave/>