

ANTEPROYECTO DEL PROYECTO/TRABAJO FIN DE GRADO

Datos del alumno

APELLIDOS: GARCÍA CUCHILLO

NOMBRE: ROBERTO

DNI:05940263F

Correo: r.gcuchillo@alumnos.upm.es

TITULACIÓN: GRADO EN INGENIERÍA DE SONIDO E IMAGEN

Datos del proyecto

DEPARTAMENTO: INGENIERÍA AUDIOVISUAL Y COMUNICACIONES

TEMA ofertado: DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR MODULAR
PARA SINTETIZADORES REALES Y VIRTUALES

TÍTULO PREVISTO EN ESPAÑOL: DISEÑO E IMPLEMENTACIÓN DE UN
CONTROLADOR MODULAR Y ESCALABLE PARA HARDWARE Y SOFTWARE MUSICAL
COMPATIBLE CON LOS PROTOCOLOS CV/GATE, MIDI Y OSC.

TÍTULO PREVISTO EN INGLÉS:

DESIGN AND IMPLEMENTATION OF A MODULAR AND SCALABLE CONTROLLER FOR
MUSICAL HARDWARE AND SOFTWARE COMPATIBLE WITH CV/GATE, MIDI, AND OSC
PROTOCOLS.

FECHA PREVISTA DEL EXAMEN:

La memoria se va a redactar en:

ESPAÑOL



INGLÉS



El proyecto/trabajo se va a defender en:

ESPAÑOL



INGLÉS



Tribunal

TUTOR: LINO GARCÍA MORALES

V.ºB.º:

SECRETARIO:

V.ºB.º:

PRESIDENTE (asignado por la SOA):

Si PFG/TFG externo:

DIRECTOR:

V.ºB.º:

Empresa/Organismo:

Financiación

Dpto.: Escuela: Otros: ALUMNO

1. INTRODUCCIÓN

El protocolo **Control Voltage/Gate (CV/Gate)** tiene sus raíces en los primeros sintetizadores modulares analógicos que emergieron en las décadas de 1960 y 1970. Cuando de la mano de entre otros, Robert Moog y Don Buchla aparecen los primeros sintetizadores modulares controlados por voltaje [1]. Este sistema de comunicación se sigue usando hoy en día en muchos sistemas de síntesis modular. Se basa en el uso de señales de voltaje controlado para modular parámetros como el tono y la modulación de los sintetizadores o para controlar el inicio y fin de las notas. Sin embargo, a pesar de su versatilidad este protocolo presenta ciertas limitaciones. Especialmente en términos de interoperabilidad entre distintos fabricantes y en la dificultad de almacenar y reproducir interpretaciones con exactitud.

El protocolo **MIDI (Musical Instrument Digital Interface)**, desarrollado entre 1981 y 1983, tras una propuesta a la *Audio Engineering Society* de Dave Smith y Chet Wood (ingenieros de Sequential Circuits) [2]. Este protocolo utiliza mensajes digitales que especifican eventos musicales, transmitiendo información sobre notas, velocidad, duración, controladores, cambios de parámetros y otros datos musicales, todo a través de señales digitales en tiempo real. No obstante, en el siglo XXI la resolución de 7 bits del protocolo y su arquitectura basada en mensajes discretos pueden ser una limitación [3], especialmente en la captura de matices expresivos y en la transmisión de datos de control en tiempo real.

Estas limitaciones no están presentes en el protocolo **OSC (Open SoundControl)**, que se basa en el envío de mensajes de control y datos en tiempo real mediante el uso de direcciones jerárquicas y argumentos tipados, transmitidos a través de UDP. Este permite comunicaciones de alta velocidad y baja latencia entre software y hardware.

Con este trabajo se pretende integrar estos tres protocolos en un único controlador, y además ofrecer una alternativa al tradicional teclado de piano que se utiliza habitualmente en los controladores comerciales.

2. OBJETIVOS

Con este proyecto se pretende implementar un controlador capaz de controlar tanto sintetizadores reales (analógicos y digitales) como instrumentos virtuales (VST).

El controlador debe tener una estructura descentralizada, que permita que los tres protocolos trabajen de forma simultánea con total interoperabilidad. Y que facilite su escalado y actualización.

También debe asegurar que el controlador tiene un manejo que permita una expresividad diferente a la que ofrecen la gran mayoría de controladores comerciales (teclados de piano, pads, etc.)

El objetivo final del proyecto es ofrecer una alternativa de control musical universal que permita evitar la barrera técnica para ciertos usuarios que puede suponer un teclado de piano. Además de reducir el coste de este tipo de controladores alternativos que se comercializan actualmente.

3. ESPECIFICACIONES Y RESTRICCIONES DE DISEÑO

- El controlador debe ser compatible con su uso en directo.
- Se debe desarrollar en un entorno de red de microcontroladores/dispositivos.
- Los módulos de los tres protocolos (CV/gate, MIDI y OSC) deben ser capaces de funcionar simultáneamente.
- El controlador debe soportar MIDI a través de conexión DIN5 y USB.
- El controlador debe proporcionar un uso intuitivo para usuarios medianamente experimentados en el uso de controladores de instrumentos electrónicos.

4. METODOLOGÍA DE TRABAJO PROPUESTA

El funcionamiento y los componentes de este proyecto pueden evolucionar durante el desarrollo de este. Por esto mismo se propone un modelo de trabajo iterativo e incremental. Que es más flexible a posibles cambios y posibilita el desarrollo de los módulos independientes de forma iterativa.

El controlador debe implementarse de forma modular, a nivel de hardware (módulo sensor y módulo controlador) y de software (módulos CV/gate, MIDI y OSC) posibilitando el desarrollo y las pruebas de cada módulo de forma independiente.

Las funcionalidades de cada módulo se deben desarrollar de forma iterativa, para garantizar la interoperabilidad entre los distintos módulos y componentes de hardware. Además, para tener flexibilidad frente a factores externos (fallos en los componentes de hardware, retrasos en envíos, etc.), este modelo es el más idóneo.

También se debe tener una validación de la experiencia de uso de las funcionalidades. Por lo que se propone una retroalimentación cíclica de pruebas de usuario. Con el fin de garantizar los objetivos de usabilidad y expresividad del controlador. Se propone enfocar estas pruebas con una metodología ágil.

5. DESGLOSE DE TAREAS Y CRONOGRAMA

Dado que el proyecto se realiza con un método de trabajo iterativo, se intercalan fases de desarrollo y fases de pruebas.

Las tareas se desglosan en la siguiente lista:

1. Estudio y planificación

- 1a Documentación y estudio previo
- 1b Requisitos y alcance de proyecto
- 1c Selección y adquisición de hardware

2. Desarrollo e implementación

- 2a Planteamiento del diseño inicial
- 2b Implementación y desarrollo del módulo controlador
- 2c Diseño, desarrollo e implementación del módulo sensor
- 2d Desarrollo módulo CV/gate
- 2e Desarrollo módulo MIDI

- 2f Desarrollo módulo OSC
- 2g Corrección de errores de integración
- 2h Implementación de mejoras de funcionamiento y funcionalidad

3. Pruebas y retroalimentación

- 3a Pruebas de funcionalidad del módulo controlador
- 3b Retroalimentación del funcionamiento del modulo sensor
- 3c Pruebas de funcionamiento del módulo CV/gate
- 3d Pruebas de funcionamiento del módulo MIDI
- 3e Pruebas de funcionamiento del módulo OSC
- 3f Pruebas de integración
- 3g Retroalimentación de funcionamiento completo
- 3h Retroalimentación de funcionalidad y manejo

4. Memoria y evaluación

- 4a Redacción de memoria del proyecto
- 4b Revisión y correcciones de memoria
- 4c Defensa del proyecto.

La cronología y planificación se detalla en el siguiente diagrama:

Tareas	Fechas estimadas		Duración (horas)	Febrero				Marzo				Abril				Mayo				Junio				Julio			
1a	06/02/2025	21/02/2025	30																								
1b	13/02/2025	28/02/2025	15																								
1c	21/02/2025	28/02/2025	10																								
2a	17/02/2025	28/02/2025	10																								
2b	03/03/2025	17/04/2025	20																								
2c	13/03/2025	27/04/2025	15																								
2d	17/03/2025	27/04/2025	30																								
2e	21/03/2025	07/05/2025	20																								
2f	21/03/2025	07/05/2025	20																								
2g	02/05/2025	11/05/2025	10																								
2h	07/05/2025	07/06/2025	25																								
3a	13/03/2025	27/03/2025	5																								
3b	17/52025	04/04/2025	5																								
3c	07/04/2025	27/04/2025	15																								
3d	13/04/2025	30/42025	10																								
3e	13/04/2025	30/42025	10																								
3f	21/04/2025	04/05/2025	10																								
3g	01/05/2025	25/05/2025	20																								
3h	13/05/2025	31/05/2025	15																								
4a	27/05/2025	10/06/2025	35																								
4b	07/06/2025	14/06/2025	5																								
4c	jul-25																										
Horas totales:			335																								

Tabla 1: diagrama de Gantt de la cronología del proyecto.

6. RECURSOS PREVISTOS

Los recursos previstos para el desarrollo del proyecto se enumeran a continuación:

Ordenador equipado con el entorno de desarrollo, además de las librerías que sean necesarias para el proyecto. Se usará un microcontrolador del ecosistema Arduino o compatible. Además del hardware externo que sea necesario.

Tanto el hardware como las librerías utilizadas pueden variar en función de las necesidades que se presenten a lo largo del desarrollo.

Para facilitar el trabajo en diferentes ordenadores se necesita hacer uso de la plataforma de control de versiones Github.com.

Como recurso de consulta, se usan papers, especificaciones técnicas, artículos de revistas, libros y manuales que sean fuentes pertinentes de información.

7. PRESUPUESTO

El presupuesto estimado se desglosa en la siguiente tabla:

HARDWARE			
Componente	Cantidad	Precio(máx.)	Comentarios
ESP32	3	10,00 €	
Force Sensitive Resistor	3	25,00 €	más envío
Soft Pot Membrane resistor 500mm	3	27,00 €	más envío
Seeduino Xiao	1	10,00 €	podría aumentar la cantidad
Teensy Ethernet adapter	1	5,00 €	
i2c LCD module	1	3,00 €	
Misc.	1	25,00 €	Boards, soldadura, cable...
Total:			229,00 €

Tabla 2: Presupuesto del proyecto

**Los gastos del proyecto corren a cargo del alumno.*

8. BIBLIOGRAFÍA

Banzi M. *Getting Started with Arduino*, 3rd ed., Make Community, LLC, 2015.

Edstrom B. *Arduino™ for Musicians: A Complete Guide to Arduino and Teensy Microcontrollers*. New York, NY: Oxford University Press, 2016.

Platt C. *Make: Electronics*, Sebastopol, CA: O'Reilly Media, Inc., 2009.

Wilson R. *Make: Analog Synthesizers*, Sebastopol, CA: Maker Media, Inc., 2013.

Wright M. y Freed A., "Open SoundControl: A New Protocol for Communicating with Sound Synthesizers," Center for New Music and Audio Technologies, University of California, Berkeley, 1997.

9. REFERENCIAS

[1] M. Jenkins, *Analog Synthesizers: Understanding, Performing, Buying: From the Legacy of Moog to Software Synthesis*, 1st ed. Oxford, UK: Focal Press, 2007.

[2] D. M. Huber, *The MIDI Manual: A Practical Guide to MIDI in the Project Studio*, 3rd ed. Burlington, MA, USA: Focal Press, 2007.

[3] B. Supper, "We hate MIDI. We love MIDI," *Focusrite Development Team*, Oct. 24, 2012. Online :
<https://focusritedevelopmentteam.wordpress.com/2012/10/24/we-hate-midi-we-love-midi/> [Última consulta 21-02-2025].

El alumno se compromete a realizar un proyecto/trabajo original, referenciando en todo caso el origen de la información utilizada, para no incurrir en casos de plagio.

Fdo.:

Aprobado por la Comisión de Ordenación Académica:

Fecha: