

# **DESARROLLO DE UN SISTEMA EMBEBIDO PARA LA ROTACIÓN DE FIBRA ÓPTICA PARA CONTROLAR LA DEPOSICIÓN UNIFORME DE MATERIAL EN UNA CÁMARA DE VACÍO**

**Alfredo González Gaviña**

Universidad de Guanajuato  
*a.gonzalezgavina@ugto.mx*

**Laureva Prouteau**

Institut Supérieur de l'Électronique et du Numérique  
*laureva@live.fr*

**Marco Bianchetti**

Universidad de Guanajuato  
*mb@ugto.mx*

**María Susana Ávila Garcia**

Universidad de Guanajuato  
*susana.avila@ugto.mx*

**Ruth Ivonne Mata Chávez**

Universidad de Guanajuato  
*ruth@ugto.mx*

## **Resumen**

Las metodologías ágiles han sido ampliamente utilizadas en los últimos años para cubrir las demandas de desarrollo de software actuales que requieren un tiempo corto de desarrollo sin sacrificar la calidad del producto. Estas metodologías dan prioridad a la satisfacción del usuario, sin embargo, no han sido tan ampliamente utilizadas en el desarrollo de software embebido ya que las características de estos proyectos difieren en varios aspectos al desarrollo de

software convencional. En este trabajo se reporta el desarrollo de un software embebido que permite controlar un motor de pasos con la finalidad de controlar la rotación de fibra óptica durante la deposición del material en una cámara de vacío, para lograr un recubrimiento uniforme y heterogéneo del material sobre la fibra. En este trabajo se presenta una metodología de trabajo que toma las características principales de la metodología Scrum, adaptándolas al desarrollo de un proyecto académico con aplicaciones en el área de ciencia de los materiales. Este proyecto reporta además los retos encontrados en el desarrollo del mismo en la colaboración multicultural y multidisciplinaria.

**Palabras Claves:** Desarrollo de software embebido, metodología ágil, Scrum.

## Abstract

*Agile methodologies have been widely used in the last decades to respond to new trends in software development that demand a faster software production without sacrificing its quality. Although these methodologies have been successful for prioritizing certain aspects of software development to ensure that user needs have been fulfilled, these have not been widely used in embedded systems development, given some fundamental differences with non-embedded software projects. In this work we report the development of embedded software for allowing the control of a step motor and the rotation of an optical fibre during the deposition of material in a vacuum camera, in order to achieve a uniform and heterogeneous deposition over the fibre. In this work the methodology used in this development is presented. This methodology adapts the Scrum methodology for a university school project with application in the research area of materials science. Finally, this project also reports on challenges of a multidisciplinary and multicultural team.*

**Keywords:** Embedded software development, Agile methodology, Scrum.

## 1. Introducción

El Departamento de Estudios Multidisciplinarios de la Sede de Yuriria de la Universidad de Guanajuato cuenta con un cuarto limpio y un Sistema de Evaporación para la Deposición de Materiales, figura 1. Este sistema al momento

tiene la capacidad de depositar un solo material por medio de evaporación térmica, pero se planea extenderlo a un sistema de deposición láser pulsado (PLD). En este trabajo se reporta el desarrollo de un sistema embebido que permite controlar un motor de pasos en dos escenarios diferentes. En el primero se pretende controlar la rotación de fibra óptica durante la deposición de un material, mientras en el segundo se pretende controlar la rotación de un blanco de uno o dos materiales para alcanzar una deposición multi-capas que se necesitará en el caso de deposición láser.



Figura 1 Sistema de evaporación térmico.

Este trabajo tiene como objetivo reportar el desarrollo de software para el control un motor de pasos que cubra estas necesidades en dos escenarios diferentes. En el primer escenario, se plantea el uso de un sistema de evaporación, Islam M. S. (2013) donde el material que se encuentra en la chalupa se evapora recubriendo así la fibra, figura 2.

En este caso se requiere girar una fibra óptica en la cámara de deposición con la finalidad de alcanzar un recubrimiento uniforme y homogéneo de la misma. Para ello se plantea el uso de un motor de pasos que permita realizar la rotación de una manera controlada, por lo que se necesita iniciar la rotación de la fibra a una velocidad definida por el usuario, y parar la rotación de la misma.

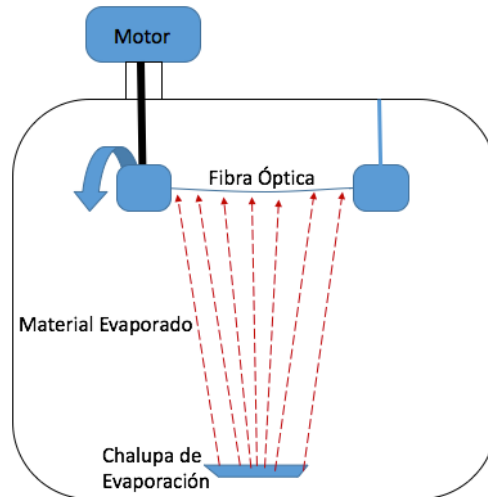


Figura 2 Diseño de Sistema de Rotación de Fibra Óptica.

En el segundo escenario se quiere controlar la rotación de un blanco de material en forma de pastilla para asistir una deposición de láser pulsado (PLD) Venkatesan T. (2014), Greer J. A. (2014), el cual se desea adquirir en un futuro, y donde la deposición del material se realiza por medio de un eje láser que incide sobre el blanco del material y lo evapora. Para este tipo de deposición se necesita un sistema que permita girar el blanco de material de tal manera que el eje láser consuma el material a lo largo de la circunferencia del blanco y no en un solo punto. En el caso más simple, y análogo al anterior, se requiere girar el contenedor del blanco de material a una velocidad definida por el usuario. El caso más complicado es cuando se quiere realizar una deposición multi-capa con dos materiales diferentes de manera secuencial. En este caso es necesario que solo un material a la vez se encuentre bajo el eje láser por un periodo de tiempo y después girar el blanco para que se realice por otro periodo de tiempo la deposición del segundo material, figura 3.

Esto se puede alcanzar posicionando la porción del blanco que se quiere evaporar bajo el eje láser y definiendo un ángulo de rotación del blanco entre el cual oscilará por un periodo de tiempo específico, de manera que el láser evapore solo el primer material. Cuando se quiere evaporar el segundo material, el motor moverá el segundo material bajo el láser y oscilará dentro de un ángulo definido

para que se evapore solo este material. El cambio de material a evaporar deberá repetirse un número predefinido de veces.

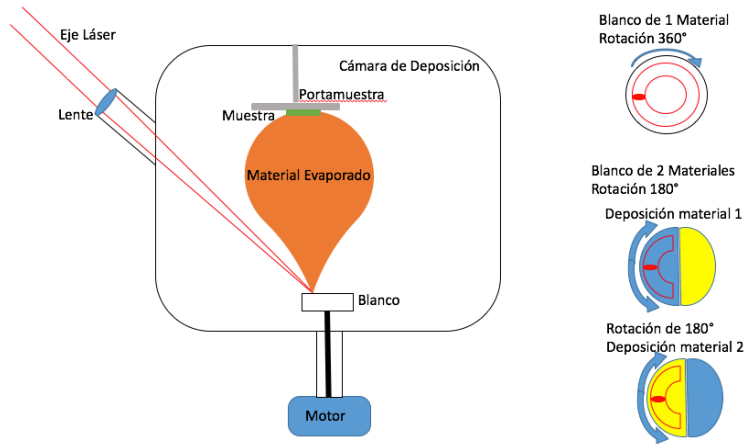


Figura 3 Deposición por PLD (izquierda) y Modos de Rotación (derecha).

En este trabajo se reporta el desarrollo de un software embebido que permita controlar un mecanismo para la rotación de fibra óptica o de blancos de material durante la deposición del mismo en la cámara de vacío. Una de las características de este desarrollo se enfoca en las diferentes disciplinas que se conjuntan para la realización del mismo y en las diferencias culturales, enfocándose específicamente al idioma nativo de los integrantes del equipo. Además, se detalla el proceso de desarrollo de software adaptado a los principios del Scrum y se reportan los retos y oportunidades que surgieron de las experiencias de este desarrollo multicultural y multidisciplinario.

### Metodologías ágiles en el desarrollo de sistemas embebidos

Los sistemas embebidos se distinguen de otros sistemas de cómputo ya que consisten de hardware especializado, software desarrollado específicamente para ese hardware y en algunas ocasiones componentes mecánicos. Los sistemas embebidos tienen que cumplir con requerimientos específicos de desempeño, tamaño del código, y tiempo de ejecución. El tamaño de código es importante ya que el hardware para el cual está desarrollado puede marcar ciertas limitantes en espacio disponible para el mismo. Los sistemas deben de cumplir con

requerimientos de tiempo de ejecución bien específicos ya que normalmente tiene que responder en tiempo real a las demandas de los sistemas en los cuales se encuentran integrados.

Este tipo de sistemas ha tenido un gran auge en los últimos años con la tendencia de "Internet de las Cosas" (IoT por sus siglas en inglés Internet of Things) en donde los artículos de uso común, como por ejemplo electrodomésticos, lentes, ropa, mochilas, etc., pueden verse extendidos con funcionalidad digital por medio de chips programables mejorando la experiencia de usuario Kranz M. (2010).

Dada que la naturaleza y características del software embebido no se han reportado una vasta cantidad de trabajos en el uso de esta metodología comparada con el desarrollo de software convencional. En Kaisti M. (2013) se reporta una investigación sobre el estado del arte en el que se plantean las siguientes preguntas de investigación:

- ¿Qué se conoce actualmente sobre los métodos ágiles en el desarrollo de sistemas y software embebido?
- ¿Son apropiados los métodos ágiles para el desarrollo de sistemas y software embebido?
- ¿Qué clase de evidencia existe para soportar estas recomendaciones?

Los resultados basados en este estudio datan de búsquedas de artículos realizadas en diciembre del 2012. De un total de 20,430 artículos obtenidos en una búsqueda inicial, los autores realizaron varios filtros lo que resultó en una selección final de 28 artículos. Los autores concluyen a partir de este análisis que uno de los factores que determinan el uso de metodologías ágiles en este tipo de desarrollo de software se refiere a la solución de problemas específicos del dominio de sistemas embebido que estas metodologías necesitan solucionar. Por ejemplo, en Ronkainen J. & Abrahamsson P. (2003) se explora la posibilidad de usar metodologías en este dominio de software y los autores proponen requerimientos para la creación de nuevas metodologías ágiles que consideren las características de este tipo de desarrollo. En este trabajo se mencionan tres características fundamentales:

- La prioridad de cubrir los requerimientos duros de tiempo real. En el desarrollo embebido normalmente existen restricciones de tiempo y desempeño, se consideran como requerimientos de tiempo real duros todos aquellos que al no alcanzarse pueden tener consecuencias severas Stankovic J. A. (1996).
- Siendo la experimentación y el desarrollo de prototipos una parte importante del desarrollo embebido la transición a código maduro y bien documentado es un reto, por lo que los autores sugieren la necesidad de técnicas que permitan incrementar de manera progresiva la madurez de un producto de software.
- El trabajo en equipos de desarrollo distribuido es una necesidad debido a las múltiples disciplinas involucradas, por lo que se propone la definición de técnicas que permitan una mejor coordinación entre estos equipos de trabajo.
- El desarrollo orientado a pruebas puede ser afectado por las limitaciones en la capacidad y espacio disponible, por lo que se requieren técnicas para construir paquetes especializados que proporcionen un ambiente adecuado y flexible.

En este trabajo se discuten ciertos aspectos que tienen relación con el segundo y tercer punto de esta lista. El equipo de desarrollo de este trabajo se encuentra integrado por miembros trabajando en diferentes áreas del conocimiento:

- Ingeniería Electrónica
- Ingeniería en Sistemas Computacionales
- Ciencia de la Materiales
- Optoelectrónica

Y con diferentes idiomas nativos como el español, italiano, y francés, se usó el inglés como lengua franca global Louhiala-Salminen L. (2012) permitiendo la comunicación entre los miembros del equipo.

## **2. Método**

Para el desarrollo de este trabajo, se siguió un marco de trabajo para la administración de procesos de desarrollo de software basado en algunos de los principios del Scrum y se definieron las siguientes etapas:

- Investigación sobre los retos del desarrollo embebido. Dada la importancia de las aplicaciones que utilizan software embebido y con el auge de nuevas tendencias como el Internet de las Cosas, se realizó una investigación en el estado del arte sobre las diferentes metodologías que soportan este tipo de desarrollo, con la finalidad de definir estrategias que permitan involucrar a especialistas de ésta área a un marco metodológico apropiado. Estas metodologías fueron presentadas en la sección anterior.
- Entrenamiento. Para los desarrollos se usó una tarjeta Arduino mega 2560, un motor de pasos 28BYJ-48 y un módulo de controlador compatible ULN2003 en la Plataforma Arduino. Dado que el equipo no se tenía previa experiencia en esta plataforma se tomó un tiempo de entrenamiento en la misma.
- Entrenamiento en los Procesos de Desarrollo de Software. Para el desarrollador en el área de electrónica, se consideró la parte de entrenamiento sobre el proceso formal de desarrollo de software embebido y sus diferentes etapas.
- Definición de necesidades de usuario. Para esto se realizó una junta para dar inicio formal al proyecto, otras reuniones se llevaron a cabo según las necesidades de desarrollo.
- Diseño conceptual del Sistema. Se definió un diseño conceptual del sistema para identificar principales actores y la relación entre los mismos, figura 4.
- Especificación de requerimientos y definición del Product Backlog. Basado en el análisis de las necesidades del usuario, se realizó una lista de características (features) del sistema a desarrollar. Se analizó esta lista, y aquellas características que fueron consideradas como complejas se rompieron en unidades que pudieran ser medibles y ejecutadas en el tiempo de periodo de la duración del sprint. Esta lista de características ordenadas



por prioridad y dependencia conformaron el Product Backlog. Diseño conceptual del Sistema. Se definió un diseño conceptual del sistema para identificar los principales actores y la relación entre los mismos. Este diseño se muestra en la figura 4.

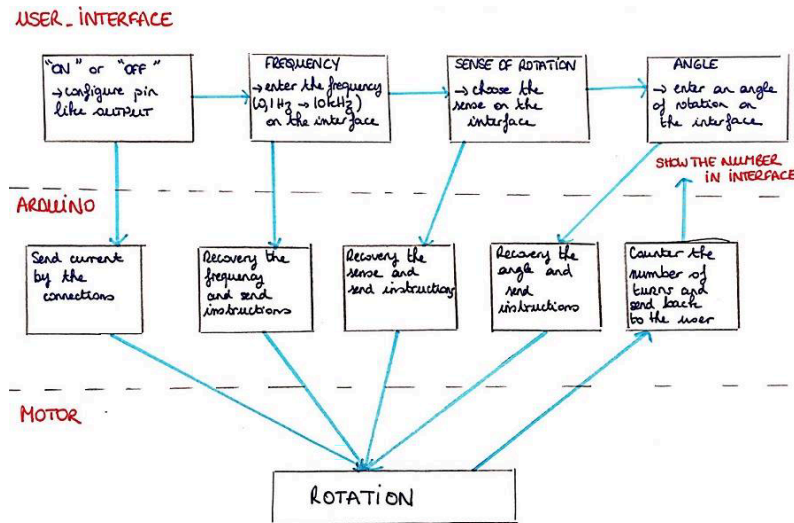


Figura 4 Diseño Conceptual del Sistema Embebido.

- Realización Iterativa de Sprints. Esta etapa incluye una Reunión de Planeación en la que se define el alcance, o *spring backlog*, y la duración del Sprint. Cabe notar que estos no tenían una duración fija y por lo que esta podía variar entre Sprints. En lugar de realizar reuniones diarias de sprint, estas se realizaban conforme fueran necesarias y podían ocurrir más de una ocasión durante el día o no ocurrir. El investigador en el área de desarrollo de software fungió como Scrum Master.
- Entrega de Código y Retroalimentación. En esta etapa se hacía la entrega del código y se proporcionaba una retroalimentación sobre la manera en que se llevó a cabo el trabajo y si hay alguna oportunidad de mejora.

### 3. Resultados

El software desarrollado fue implementado en Arduino y con una interfaz de usuario en la computadora desarrollada en Visual C++. El sistema es capaz de realizar las siguientes acciones:

- Encendido y apagado del motor.
- Especificación de velocidad de giro del motor y de la fibra óptica.
- Sentido de rotación del motor, para indicar si será en el sentido de las manecillas del reloj o en contra.
- Limitación de rotación de ángulo y cambio de secuencia de rotación.

Esta característica fue definida para controlar la rotación de los materiales a depositar y no de la fibra óptica en sí.

Además, se cuidó que el movimiento del motor fuera suave para evitar movimientos bruscos que pudieran potencialmente dañar la fibra óptica. La funcionalidad desarrollada servirá como base para la implementación del sistema que permita la rotación, no sólo de la fibra óptica, sino también de los blancos al momento de realizar una deposición con PLD. El sistema necesita otros elementos que aún no han sido diseñados, estos incluyen, pero no se limitan a:

- Sistema mecánico de montaje de la fibra óptica.
- Sistema mecánico de montaje de los blancos en la cámara de deposición.

Por otro lado, este trabajo reporta en los diferentes retos del desarrollo de software embebido usando la metodología planteada en este trabajo enfocándose principalmente en los aspectos que fueron nuevos para el desarrollador del área de electrónica:

- Aprendizaje de los Procesos de Desarrollo de Software. Se observó una ligera resistencia por parte del estudiante para seguir los procesos de desarrollo de software, ya que no se tenía el hábito. El primer instinto fue el de comenzar a programar la aplicación de manera directa en la computadora. Se indicaba preferencia por las actividades de desarrollo de software.
- Identificación de necesidades de usuario. En esta etapa se hicieron presentes los retos en cuanto a la comunicación entre ellos se encuentra el entender la naturaleza del problema, o el dominio del tema en el cual se desarrollará el proyecto. Siendo el área de materiales, y en específico la

deposición de los mismos, nueva para el estudiante se tuvo que aprender terminología que permitió con el paso del tiempo adquirir un lenguaje común de comunicación. Otro de los aspectos a considerar en este rubro se refiere la comunicación de los que un sistema embebido debe realizar. La explicación de la manera en que los motores girarían los diferentes elementos del sistema causaban confusión por que se emplearon elementos visuales para asistir la explicación y funcionamiento deseado.

- Definición del Backlog. Siguiendo las prácticas del Scrum se tenía que definir el Backlog de características del sistema a desarrollar. Esta especificación que, aunque muy concisa, fue catalogada como parte de un proceso de documentación.
- Diseño Conceptual. El diseño conceptual también fue un concepto nuevo y la identificación de la interface entre los diferentes elementos y su representación de manera esquemática representó un reto en la comunicación y especificación de lo que se requería.
- Desarrollo incremental, en este trabajo se definieron sprints que permitieran cubrir dos aspectos:
  - ✓ El aprendizaje de las herramientas y tecnologías utilizadas.
  - ✓ Que los productos generados fueran funcionales y permitieran una satisfacción continua de las necesidades del cliente.

#### **4. Discusión**

Como se especifica en la literatura investigada, el uso de metodologías ágiles para asistir el desarrollo de sistemas embebidos necesita ser analizado y las metodologías modificadas. Los retos encontrados en este ámbito del desarrollo de software y en este proyecto fueron primordialmente en aspectos de comunicación y adaptación al proceso de desarrollo de software.

En cuanto a la comunicación, intervienen varios factores entre los que se encuentra el uso de terminología técnica especializada. El contexto de aplicación del proyecto estaba enfocado en el área de materiales y de optoelectrónica, el desarrollador tiene formación en el área de ingeniería electrónica, y el Scrum

Master formación como ingeniero de software. Cada uno de los involucrados aprendió nuevos conceptos de otras áreas, pero este proceso tomó tiempo para que el equipo pudiera hablar un lenguaje en común, usando el inglés como *Lingua Franca*.

En cuanto a la adaptación al proceso de desarrollo de software, cabe la pena mencionar varios aspectos. El desarrollo de software embebido no es exclusivo de especialistas en el área de sistemas computacionales, sino también de especialistas en las áreas de electrónica, mecatrónica, etc., de tal manera que se debe buscar la preparación de estos especialistas en el uso de metodologías de software para fomentar el trabajo colaborativo, y multidisciplinario en el desarrollo no solo de software sino de sistemas embebidos completos.

El seguimiento estricto de metodologías de desarrollo ágiles en contextos académicos puede ser complicado por las limitantes de tiempo y recursos con los que se cuentan. La metodología usada adaptó algunos de los principios de la metodología Scrum, en la que se consideraron los siguientes aspectos: definición del rol de Scrum Master, análisis de necesidades de usuario y constante comunicación con el mismo, definición de un backlog y de sprints. Para definir el backlog del Producto se realizó un análisis de las características del software y de la definición de interdependencia entre ellas. Además, para la definición del sprint backlog se consideraron las prioridades del usuario lo que permitió que los productos en las diferentes etapas pudieran realizarse de manera incremental. En sistemas embebidos más complejos, esto puede no ser posible. La comunicación constante con el usuario y la flexibilidad de recibir nuevos requerimientos permitió el mantener la satisfacción del usuario durante el proceso de desarrollo.

No era posible llevar a cabo reuniones diarias de Scrum por lo que estas se realizaban a medida que se iban necesitando. Al final de cada sprint se realizaba una reflexión sobre lo aprendido, las dificultades encontradas, y la manera de retroalimentar estas experiencias en el siguiente sprint. Estas dificultades encontradas estaban directamente relacionadas con los retos mencionados en esta sección.

Por otro lado, en este trabajo no se tuvieron restricciones importantes en cuanto al tamaño del software, como se podría tener en un ambiente de aplicaciones reales en sistemas embebidos más complejos. Los módulos desarrollados en este trabajo se usarán para orquestar soluciones más complejas y para ello se tendrán que contemplar pruebas para analizar la respuesta de estas soluciones en tiempo real.

El sistema de rotación de la fibra tiene retos importantes en cuanto al diseño del sistema mecánico el cual deberá tener en consideración la fragilidad de la fibra para su fijación y rotación dentro de la cámara de vacío. Este diseño es parte del trabajo futuro.

## **5. Bibliografía y Referencias**

- [1] Beck K. , Beedle M. , van Bennekum A. , Cockburn A. , Cunningham,W. ,Fowler M. Grenning J. , Highsmith J. , Hunt A. , Jeffries R. , Kern J. , Marick B. , Martin R. , Mellor S. , Schwaber K. , Sutherland J. , Thomas D. (2001). Agile manifesto. <http://agilemanifesto.org/>.
- [2] Dingsø T., Nerur S., Balijepally V., & Moe N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *J. Syst. Softw.*, vol. 85, (no. 6), pp. 1213-1221.
- [3] Greene B, (2004). Agile methods applied to embedded firmware development. *Proc. Agil. Dev. Conf. ADC 2004*, pp. 71-77.
- [4] Greer J. A. (2014). History and current status of commercial pulsed laser deposition equipment. *J. Phys. D Appl. Phys. J. Phys. D Appl. Phys. J. Phys. D Appl. Phys*, vol. 47, (no. 47), pp. 34005-10.
- [5] Islam M. S., Hossain M. F., Shaalan N. M., & Ali M. M. (2013). Fabrication of Nanostructured SnO<sub>2</sub> Thin Films by A Simplified Thermal Evaporation System, vol. 1, (no. 1), pp. 120-125.
- [6] Kaisti M., Rantala V., Mujunen T., Hyrynsalmi S., Könnölä K., Mäkilä T., & Lehtonen T. (2013). Agile methods for embedded systems development – a literature review and a mapping study. *EURASIP J. Embed. Syst.*, vol. 2013, (no. 1), p. 15.

- [7] Kettunen P. Laanti M. (2008). Combining agile software projects and large-scale organizational agility. *Softw. Process Improv. Pract.*, vol. 13, (no. 2), pp. 183-193.
- [8] Könnölä K., Suomi S., Mäkilä T., Jokela T., Rantala V., & Lehtonen T. (2016). Agile methods in embedded system development: Multiple-case study of three industrial cases. *J. Syst. Softw.*, vol. 118, pp. 134-150.
- [9] Kranz M., Holleis P., & Schmidt A., (2010). Embedded interaction: Interacting with the internet of things. *IEEE Internet Comput.*, vol. 14, (no. 2), pp. 46-53.
- [10] Louhiala-Salminen L. & Kankaanranta A., (2012). Language as an issue in international internal communication: English or local language? If English, what English? *Public Relat. Rev.*, vol. 38, (no. 2), pp. 262-269.
- [11] Ronkainen, J., Abrahamsson, P. (2003). Software development under stringent hardware constraints: do agile methods have a chance? *Extrem. Program. Agil. Process. Softw. Eng.*, pp. 1012-1012.
- [12] Stankovic J. A., (1996). Real-time And Embedded Systems. *ACM Comput. Surv.*, vol. 28, (no. 1), pp. 205-208.
- [13] Venkatesan T., (2014). Pulsed laser deposition, invention or discovery? *J. Phys. D. Appl. Phys.*, vol. 47, (no. 3), p. 034001.