



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Desarrollo de Plataforma para impresión remota 3D

Grado en Ingeniería Mecánica

ALUMNO: Alejandro Carnicero Solanas

DIRECTORES: Andrés Suarez García
Carlos Casqueiro Placer

CURSO ACADÉMICO: 2016-2017

Universida_{de}Vigo



**Centro Universitario de la Defensa
en la Escuela Naval Militar**

TRABAJO FIN DE GRADO

Desarrollo de Plataforma para impresión remota 3D

Grado en Ingeniería Mecánica
Intensificación en Tecnología Naval
Cuerpo General

Universidad de Vigo

RESUMEN

El mundo de la impresión 3D está en pleno crecimiento: no en vano, una parte muy importante del progreso realizado en este aspecto es gracias a una extensa comunidad de desarrolladores. En este ámbito existen tareas en constante evolución para convertir el proceso en una tarea lo más automatizada posible.

El presente proyecto busca desarrollar una plataforma de impresión 3D remota que pueda ser instalada de forma sencilla en cualquier impresora 3D, empleando una placa Raspberry Pi para el control remoto de la impresora y un microcontrolador Arduino para las funciones de control y monitorización de la misma. La misión de ambos módulos, funcionando de manera integrada, es que cualquier impresora 3D doméstica pueda funcionar de manera remota y con un alto grado de autonomía de su operador, minimizando los tiempos de manipulación *in situ* de la impresora.

A la culminación del proyecto, se ha llegado a una plataforma remota de impresión 3D que puede funcionar de manera remota y adicionalmente, incluye una serie de automatismos y medidas de seguridad pasivas que convierten a dicha plataforma en un servicio con alto grado de autonomía. Además, se proponen nuevas líneas de investigación para conseguir una plataforma lo más completa posible, en este aspecto se plantea la introducción de nuevas tecnologías aplicadas a la impresión 3D además de la integración de la plataforma propuesta en las Fuerzas Armadas.

PALABRAS CLAVE

Raspberry Pi, Arduino, impresora 3D, control remoto

AGRADECIMIENTOS

A mis padres, Félix y Rosa,
artífices de que hoy pueda estar entregando este proyecto,
a pesar de las dificultades,
poniendo fin a mi etapa formativa como alumno de la Escuela Naval Militar.

A mi novia, Elena
por haberme apoyado en todas mis decisiones
así como ser mi fuente de buen humor y optimismo día tras día

En primer lugar, me gustaría agradecer el trabajo realizado por mis tutores,
Andrés Suarez y Carlos Casqueiro
por su motivación, ánimos y apoyo constante por su parte en la realización de este proyecto.

Al Centro Universitario de la Defensa de Zaragoza,
el cual me facilitó información sobre diversos proyectos relacionados.
Por su colaboración y ánimo.

CONTENIDO

Contenido	1
Índice de Figuras	5
Índice de Tablas	9
Glosario	10
1 Introducción y objetivos	13
1.1 Introducción y motivación	13
1.1.1 Antecedentes de la impresión 3D remota	13
1.1.2 Usos de la impresión 3D en el ámbito de la educación	13
1.1.3 Usos de la impresión 3D remota en el ámbito de las Fuerzas Armadas	14
1.1.4 Motivación	14
1.2 Objetivos	14
1.3 Recursos Utilizados	15
1.3.1 Hardware	15
1.3.2 Software	15
1.4 Estructura del proyecto	15
1.5 Estructura de la memoria	16
2 Estado del arte	17
2.1 Historia de la impresión 3D	17
2.1.1 Introducción	17
2.1.2 Evolución de las impresoras 3D	17
2.1.3 Impresoras 3D industriales	18
2.1.4 Técnicas de impresión 3D en la Industria	19
2.1.5 Impresoras 3D domésticas	21
2.1.6 Técnicas de impresión 3D en el entorno doméstico	22
2.2 Estado Actual de la impresión 3D	24
2.2.1 Introducción	24
2.2.2 Efectos sobre la estructura de producción	24
2.2.3 Impresión 3D aplicada al ámbito militar	25
2.3 Impresión 3D de manera remota	27
2.3.1 Introducción	27
2.3.2 Octoprint	27
2.3.3 Astroprint	29
2.3.4 Conclusiones de los servicios de impresión de manera remota	30
2.4 Raspberry Pi	30

2.4.1	Introducción.....	30
2.4.2	Historia	31
2.4.3	Estado actual y futuro de la plataforma	32
2.5	Impresión 3D basada en Raspberry Pi	33
2.5.1	Introducción.....	33
2.6	Arduino	33
2.6.1	Historia de Arduino	33
2.6.2	Descripción de la plataforma.....	34
2.6.3	Conclusión.....	34
3	Desarrollo del TFG.....	35
3.1	Descripción de la arquitectura de una plataforma remota de impresión 3D.....	35
3.1.1	Introducción y características del sistema	35
3.1.2	Diseño del sistema de impresión remota.....	35
3.1.3	Descripción del sistema de impresión 3D remota	36
3.1.4	Descripción de la interfaz de red del sistema de impresión 3D remota	38
3.2	Elección del servicio de impresión 3D	40
3.3	Descripción del <i>software</i> Octoprint y de su interfaz web	41
3.3.1	Descripción del programa Octoprint	41
3.3.2	Descripción de la interfaz web Octoprint.....	42
3.3.3	Descripción de las opciones de configuración de la interfaz web de Octoprint.....	46
3.4	Instalación y configuración de Octoprint.....	49
3.4.1	Instalación de Octoprint.....	49
3.4.2	Configuración de Octoprint	52
3.4.3	Configuración de la interfaz web de Octoprint	54
3.5	Creación de un bot de Telegram	57
3.5.1	Descripción del bot de Telegram creado.....	57
3.5.2	Creación del bot de Telegram.....	58
3.5.3	Instalación y configuración del plugin de Telegram en Octoprint	60
3.5.4	Valor añadido y conclusión	61
3.6	Seguridad de la plataforma remota de impresión 3D.....	61
3.7	Desarrollo de un sistema auxiliar para la plataforma de impresión 3D.....	62
3.7.1	Introducción y características de la plataforma de sistemas auxiliares	62
3.7.2	Requisitos de la placa controladora del sistema auxiliar	63
3.7.3	Diseño del sistema auxiliar de la plataforma de impresión 3D	63
3.7.4	Descripción de los componentes del sistema auxiliar de impresión 3D.....	64
3.7.5	Descripción del montaje del sistema auxiliar de la plataforma de impresión 3D	67

3.7.6 Sistema de acondicionamiento de temperatura del espacio de impresión.....	68
3.7.7 Sistema de acondicionamiento lumínico del entorno de impresión.	72
3.7.8 Detector de falta de filamento en la impresora 3D.....	75
3.7.9 Software general del sistema.....	80
3.7.10 Detector de sobre-intensidades en los motores de la impresora 3D.....	81
4 Resultados / Validación / Prueba.....	83
4.1 Primeras pruebas del sistema Octoprint.....	83
4.1.1 Conexión entre el ordenador y la Raspberry Pi.....	83
4.1.2 Conexión de la impresora a la Raspberry Pi.....	84
4.1.3 Verificaciones en el funcionamiento de Octoprint.....	84
4.1.4 Carga de los archivos a Octoprint.....	86
4.1.5 Primera impresión utilizando Octoprint.....	87
4.2 Validación del sistema.....	87
4.2.1 Preparación de la pieza.....	87
4.3 Pruebas y validación del <i>bot</i> de Telegram.....	89
4.3.1 Primeras pruebas con el bot de Telegram.....	89
4.3.2 Validación del sistema.....	91
4.4 Pruebas y validación del sistema auxiliar de impresión 3D.....	91
4.4.1 Verificación de la instalación del sistema.....	91
4.4.2 Verificación del sistema de acondicionamiento lumínico.....	92
4.4.3 Verificación del detector de falta de filamento.....	93
4.4.4 Verificación de sistema de acondicionamiento de la impresora.....	93
5 Conclusiones y líneas futuras.....	96
5.1 Conclusiones del presente proyecto.....	96
5.2 Opinión personal.....	97
5.3 Continuación del presente proyecto.....	97
5.4 Líneas futuras de estudio.....	97
5.4.1 Integración de la plataforma con una aplicación para Android.....	97
5.4.2 Integración de un sistema de luz Ultravioleta para sintetizar el material de impresión... ..	97
5.4.3 Integración de la plataforma remota de impresión 3D en la red intranet de la Armada... ..	98
6 Bibliografía.....	99
Anexo I: Especificaciones técnicas.....	104
Especificaciones técnicas de la impresora 3D utilizada en el proyecto.....	104
Especificaciones técnicas de los componentes de la plataforma de impresión 3D remota.....	104
Especificaciones técnicas de los componentes del sistema auxiliar.....	106

Anexo II: Código del sistema auxiliar.....108

ÍNDICE DE FIGURAS

Figura 1-1 Maqueta realizada mediante impresión 3D (Thingiverse, 2014)	13
Figura 1-2 Impresora 3D a bordo de un buque del U.S. Essex (Stinson, 2014)	14
Figura 1-3 Topología básica de la plataforma de impresión 3D remota	16
Figura 2-1 Primer prototipo de la impresora FDM "ReRap" (Makerbot, 2017).....	18
Figura 2-2 Boeing A350 (Airbus, 2017)	19
Figura 2-3 Corazón humano impreso en 3D (Stratasys, 2015)	19
Figura 2-4 Esquema del proceso de impresión SLA (3D Systems, 2017).....	20
Figura 2-5 Esquema del proceso de impresión SLS (Deckard, 1986)	21
Figura 2-6 Principales empresas de fabricación de impresoras 3D (3DHubs, 2017)	22
Figura 2-7 Esquema del proceso de impresión FDM (Palm, 2008).....	22
Figura 2-8 Figura de chocolate impresa en 3D (technabob, 2017)	23
Figura 2-9 Escultura tridimensional realizada con un "bolígrafo 3D" (Wobbleworks, 2017).....	23
Figura 2-10 Piezas de Lego mediante diseño CAD para su impresión (Thingiverse, 2014)	24
Figura 2-11 Pistoleta de un fusil M16 fabricado por impresión 3D (Impresióntresde, 2016)	25
Figura 2-12 Impresión 3D a bordo del B.I.O Hespérides (La catedral del Mar, 2017)	26
Figura 2-13 Caja de un <i>logger</i> fabricada mediante impresión 3D a bordo	26
Figura 2-14 Reparación de un equipo con una herramienta impresa en 3D (Pacheco, 2014)	27
Figura 2-15 Logo de Octoprint (OctoPrint)	28
Figura 2-16 Interfaz web de Octoprint	28
Figura 2-17 Logo de Astroprint (Astroprint, 2017)	29
Figura 2-18 Interfaz web de Astroprint (Astroprint, 2017).....	30
Figura 2-19 Logo de <i>Raspberry Pi Foundation</i> (Raspberry Pi Foundation)	31
Figura 2-20 Comparación de la Raspberry Pi Model A y model B (Raspberry Pi Foundation)	31
Figura 2-21 Módulo Astro-Pi a bordo de la Estación Espacial Internacional (EEI) (NASA, 2016).....	32
Figura 2-22 Modelo "Zero" de Raspberry Pi (Raspberry Pi Foundation).....	33
Figura 2-23 Logo de <i>Arduino Foundation</i> (Arduino Foundation).....	34
Figura 3-1 Topología de la plataforma de impresión remota en 3D	36
Figura 3-2 Impresora 3D Witbox I (Bq, 2013)	37
Figura 3-3 Cámara web Logitech C170	37
Figura 3-4 Montaje del router en la plataforma de impresión remota	38
Figura 3-5 Montaje del sistema en la impresora 3D	38
Figura 3-6 Descripción de la interfaz de red del sistema de impresión remota	39
Figura 3-7 Respuesta al comando <i>ifconfig</i> en la <i>shell</i> de Raspbian	39
Figura 3-8 Comparativa entre las interfaces de <i>Octoprint</i> y <i>Astroprint</i>	40

Figura 3-9 Menú de configuración de Octoprint.....	41
Figura 3-10 Menú del apartado de configuración <i>Advanced Options</i>	42
Figura 3-11 Interfaz web de Octoprint.....	43
Figura 3-12 Apartado <i>Control</i> de Octoprint.....	44
Figura 3-13 Apartado <i>GCode Viewer</i> de Octoprint	44
Figura 3-14 Apartado <i>Terminal</i> de Octoprint	45
Figura 3-15 Apartado <i>Timelapse</i> de Octoprint.....	45
Figura 3-16 Ventana de configuración de Octoprint.....	46
Figura 3-17 Menú desplegable de configuración de los perfiles de impresión.....	47
Figura 3-18 Perfiles de impresión de CuraEngine	47
Figura 3-19 Comparativa de dos piezas con distintas alturas de capa (Imprimalia, 2016).....	48
Figura 3-20 Pieza con imperfecciones producto de no utilizar retracción (Imprimalia, 2016)	48
Figura 3-21 Comparativa de piezas impresas con distinta densidad (Imprimalia, 2016)	48
Figura 3-22 Apartado webcam de la configuración del sistema	49
Figura 3-23 Página web oficial de Octoprint (OctoPrint).....	50
Figura 3-24 Software para formatear dispositivos de almacenamiento (<i>SDFormateer</i>)	50
Figura 3-25 Software para grabar imágenes en una tarjeta SD (<i>ApplePi-Baker</i>)	51
Figura 3-26 Documento de texto donde se almacenan los parámetros de red de Octoprint.....	51
Figura 3-27 Interfaz de usuario de <i>Raspbian</i>	52
Figura 3-28 Menú de configuración de Octoprint.....	53
Figura 3-29 Panel de configuración del teclado en Octoprint.....	53
Figura 3-30 Submenú " <i>Configuración avanzada</i> "	54
Figura 3-31 Entorno gráfico de Octoprint.....	54
Figura 3-32 Cuadro de conexión con la impresora 3D en Octoprint	55
Figura 3-33 Apartado <i>Control</i> de la interfaz web	55
Figura 3-34 Botón <i>home</i>	55
Figura 3-35 Pestaña <i>Settings</i> de Octoprint	56
Figura 3-36 Apartado de configuración de temperaturas de impresión en Octoprint.....	56
Figura 3-37 Ventana del Plugin Manager de Octoprint	57
Figura 3-38 Logo de Telegram Messenger (Telegram messenger, 2017)	57
Figura 3-39 Búsqueda del <i>bot</i> "CudPrinter"en Telegram	58
Figura 3-40 Presentación inicial de "botfather" en Telegram	59
Figura 3-41 Devolución del <i>token</i> de acceso del <i>bot</i>	59
Figura 3-42 Envío de comandos al bot de Telegram	60
Figura 3-43 Configuración del plugin de Telegram para Octoprint.....	60
Figura 3-44 Permisos de usuario sobre el <i>bot</i> desde la aplicación Telegram	61

Figura 3-45 Control de acceso del software Octoprint	61
Figura 3-46 Permisos de visualización del <i>bot</i>	62
Figura 3-47 Permisos de acción del <i>bot</i>	62
Figura 3-48 Descripción del <i>hardware</i> utilizado en el sistema auxiliar	64
Figura 3-49 Partes de la placa <i>Arduino Uno</i> (Arduino Foundation).....	64
Figura 3-50 Sensor de temperatura digital	65
Figura 3-51 Tira de LEDs utilizada en el sistema.....	66
Figura 3-52 Relé tipo "keyes"	66
Figura 3-53 Ventilador de hasta 12V	66
Figura 3-54 Foto-resistor.....	67
Figura 3-55 Topología de las conexiones del sistema auxiliar de la plataforma de impresión 3D..	67
Figura 3-56 Pieza con un fallo de " <i>warping</i> " (Imprimalia, 2016).....	68
Figura 3-57 Piezas en voladizo impresas sin refrigeración (Imprimalia, 2016)	69
Figura 3-58 Componentes del sistema de acondicionamiento del sistema de impresión	69
Figura 3-59 Esquema del montaje del sistema de acondicionamiento de temperatura.....	70
Figura 3-60 IDE de Arduino durante la programación del sistema de acondicionamiento.	70
Figura 3-61 Diagrama de flujo del sistema de climatización de la plataforma de impresión	71
Figura 3-62 Componentes del sistema de acondicionamiento lumínico del entorno de impresión. 72	
Figura 3-63 Esquema del montaje del sistema de acondicionamiento lumínico	73
Figura 3-64 Entorno de programación del sistema de acondicionamiento lumínico	73
Figura 3-65 Diagrama de flujo del acondicionamiento lumínico de la plataforma de impresión... 74	
Figura 3-66 Detalle del puerto serie de la IDE de Arduino	75
Figura 3-67 Diseño inicial del detector de filamento.....	76
Figura 3-68 Ensamblaje de diseño funcional del detector de filamento	76
Figura 3-69 Primer prototipo de detector de filamento montado.....	77
Figura 3-70 Piezas del detector de falta de filamento	77
Figura 3-71 Diseño final del detector de filamento.....	78
Figura 3-72 Detalle de las pestañas para la sujeción de la tapa	78
Figura 3-73 Diagrama de flujo del detector de falta de filamento	79
Figura 3-74 Esquema del montaje definitivo del detector de filamento de la impresora 3D.....	80
Figura 3-75 Esquema de la placa controladora de la impresora 3D.....	81
Figura 3-76 Driver A4988 de control de los motores paso a paso.....	82
Figura 4-1 Detalle del router inalámbrico en la ubicación actual de la impresora.....	83
Figura 4-2 Interfaz de ajuste de temperatura del extrusor en Octoprint.....	84
Figura 4-3 Instalación temporal de la cámara web en la impresora 3D.....	85
Figura 4-4 Apartado control con <i>streaming</i> de una impresión en curso	85

Figura 4-5 Previsualización de fabricación del código <i>gcode</i>	86
Figura 4-6 Fotograma de uno de los <i>timelapses</i> realizados	86
Figura 4-7 <i>Izquierda</i> : Previsualización en Octoprint. <i>Derecha</i> : Producto en impresión	87
Figura 4-8 Posicionamiento de la probeta plana en la impresora 3D.....	88
Figura 4-9 Detalle de los parámetros de impresión utilizados	88
Figura 4-10 Pieza final impresa con una densidad del 50%	89
Figura 4-11 Respuesta del <i>bot</i> al comando <i>/status</i>	90
Figura 4-12 Respuesta al comando <i>/con</i> del <i>bot</i>	90
Figura 4-13 Carga y visualización de los archivos disponibles en Octoprint	90
Figura 4-14 Opciones al recibir una respuesta al comando <i>/print</i>	91
Figura 4-15 Montaje provisional de los periféricos sobre la impresora 3D.....	91
Figura 4-16 Lecturas leídas por el sensor durante las pruebas del sistema.....	92
Figura 4-17 Comparativa de la imagen recibida con luz auxiliar	92
Figura 4-18 Pieza final del contacto del resorte.....	93
Figura 4-19 Lectura del sensor de temperatura en Arduino.....	93
Figura A-1 Esquema de conexiones de la Raspberry Pi	104
Figura A-2 Esquema de los puertos GPIO de la Raspberry Pi Model B	105
Figura A-3 Esquema de conexión del sensor de temperatura	106
Figura A-4 Esquema del circuito del relé tipo Keyes (Microjpm, 2017).....	107

ÍNDICE DE TABLAS

Tabla 1 Comparativa de los modelos disponibles de Raspberry Pi	32
Tabla 2 Descripción de las opciones de configuración de Octoprint.....	41
Tabla 3 Descripción del sub-menú <i>Advanced Options</i>	42
Tabla 4 Descripción de las opciones de configuración de la cámara web	49
Tabla 5 Comandos del bot de Telegram.....	58

GLOSARIO

- ABS** (Acrilonitrilo butadieno estireno) Polímero utilizado, entre otras cosas, para la fabricación aditiva mediante impresión 3D.
- ACK** Abreviatura de *Acknowledge*, típico mensaje de control en redes de ordenadores.
- API** (*Application Programming Interface*) Conjunto de aplicaciones y subrutinas que un desarrollador expone públicamente para poder comunicar distintos componentes de software.
- CAD** (*Computer-Aided Design*) Herramientas computacionales para el diseño asistido por ordenador.
- CuraEngine** Motor de impresión más ampliamente utilizado en el mundo de la impresión 3D doméstica.
- Fork:** Término en el ámbito del desarrollo de *software* para hacer referencia a un sistema basado en un código fuente determinado para tomar una dirección distinta al proyecto inicial que añada o cambie capacidades.
- FPS:** (*Frames Per Second*) Fotogramas por segundo. Uno de los parámetros principales de un video.
- IDE** (*Integrated Development Environment*) Aplicación informática que proporciona servicios que facilitan la tarea del desarrollador.
- LAN** (*Local Area Network*) Red de área local. Conjunto de ordenadores conectados en red que no tienen por qué tener acceso a Internet.

Motor de impresión *Software* generalmente utilizado por las impresoras 3D del tipo FDM para generar las capas que más tarde extruirá la impresora.

NC (*Normally closed*) Circuito generalmente cerrado de un relé.

NO (*Normally Open*) Circuito generalmente abierto de un relé.

Opensource Término para referirse generalmente a proyectos de código libre no lucrativos.

PLA Ácido Poliláctico. Polímero ampliamente utilizado en impresión 3D.

Plugin Aplicación que se relaciona con otra generalmente de mayor entidad para darle una funcionalidad extra o mejorarla.

PWM (*Pulse-Width Modulation*) Modulación de pulso para modificar la frecuencia de trabajo de una señal. Utilizada generalmente para variar la cantidad de energía enviada a través de una señal digital.

Shell Interfaz de usuario más básica para acceder a las funcionalidades del Sistema Operativo.

SO (Sistema Operativo). Relativo a la informática

SSH (*Secure SHell*) Protocolo de red para acceder a máquinas remotas a través de una red.

String Elemento utilizado en programación que hace referencia a una cadena de caracteres.

WAN (*Wide Area Network*). Referido a Internet cuando es mencionado

1 INTRODUCCIÓN Y OBJETIVOS

1.1 Introducción y motivación

1.1.1 Antecedentes de la impresión 3D remota

La impresión 3D es una realidad desde la década de los ochenta, aunque su abaratamiento y producción en masa, le ha hecho alcanzar día a día una posición preferente en los hogares de todo el mundo. Sin embargo, no debemos olvidar todavía que es una tecnología en desarrollo. En un mundo conectado a internet, resulta llamativo como la impresión 3D, a día de hoy, aún se asoma tímidamente a las redes. Las plataformas de impresión 3D actuales, en su mayoría *hardware* de código abierto, están muy lejos de integrar todas las capacidades que potencialmente podrían contener, sin embargo, el actual concepto del *Internet de las Cosas* (IOT) está cada vez más extendido y esto incluye también, de alguna manera, a la impresión 3D.

1.1.2 Usos de la impresión 3D en el ámbito de la educación.

El entorno de la educación es esencial para el desarrollo de las tecnologías emergentes como la impresión 3D. Los nuevos modelos de enseñanza incluyen cada vez más el uso de estas tecnologías en el aprendizaje, la investigación y la gestión de la información. La impresión de mapas topográficos, el diseño arquitectónico como se puede observar en la Figura 1-1 o el estudio de la anatomía en el ámbito de la biología son solo algunos de los ejemplos en los que la impresión 3D supone un recurso educativo de alto valor. Todo esto, sumado a la capacidad de operar estas impresoras de manera remota, desde cualquier lugar a través de la red, alcanza una nueva dimensión en términos de inmediatez y personalización.

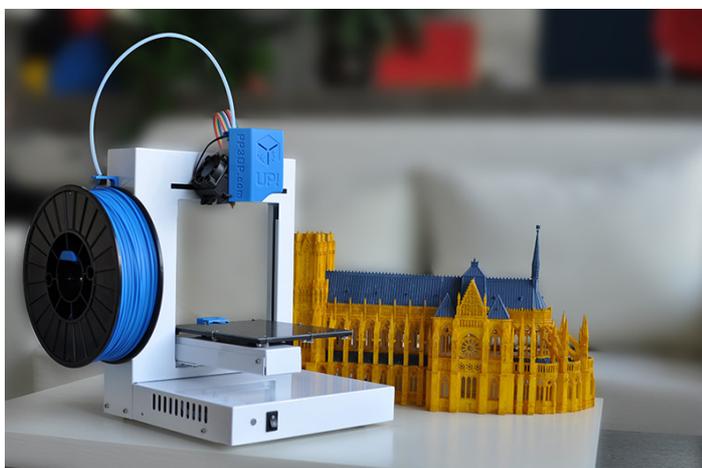


Figura 1-1 Maqueta realizada mediante impresión 3D [1]

1.1.3 Usos de la impresión 3D remota en el ámbito de las Fuerzas Armadas

Es bien sabido que, habitualmente, el sector de Defensa está a la vanguardia en el uso de nuevas tecnologías. La superioridad tecnológica se ha convertido en el pasado reciente en un factor decisivo en un conflicto; un ejemplo de esto es la carrera espacial, la implantación de satélites o incluso la bomba atómica. En los últimos años, la crisis económica que ha acuciado a Europa se ha traducido en recortes drásticos en el sector público. Esta es una de las causas principales que explican el por qué algunas Fuerzas Armadas, como la española, todavía se encuentran en una situación muy conservadora respecto a la implantación de nuevas tecnologías, como la impresión 3D.

Sin embargo, en los últimos años su implantación está tomando un ritmo lento pero seguro y en un entorno de continua rotación e inestabilidad, la adopción de tecnologías para operar estas impresoras de manera remota, se convierten en un requisito casi imprescindible. De esta manera, con una ligera supervisión y un mínimo mantenimiento, cada vez más, va a resultar una realidad el poder imprimir una pieza en alta mar o en una base avanzada en el extranjero durante una misión. Este hecho está llamado revolucionar algunos ámbitos como el apoyo logístico o el mantenimiento de las Unidades. Un ejemplo claro es el USS Essex (Figura 1-2) el cual ya incorpora tecnología de impresión 3D a bordo para la fabricación de piezas para diversos propósitos.



Figura 1-2 Impresora 3D a bordo de un buque del U.S. Essex [2]

1.1.4 Motivación

El objetivo principal de este proyecto responde a una estructura básica de problema-solución. Los objetivos nacen de una necesidad real que existe en el Centro Universitario de la Defensa de Marín para separar el espacio de trabajo de la impresora 3D de los despachos del profesorado. A este respecto, se había de trabajar en salvar de la manera más eficiente posible los posibles inconvenientes de esta solución asociados a la distancia de trabajo y a la reducción drástica de los tiempos de manipulación *in-situ* de la máquina. Además, debía de maximizarse la capacidad de acceso de la impresora por parte del mayor número de personas posibles manteniendo la seguridad en el uso de la misma.

1.2 Objetivos

El objetivo inicial definido para este proyecto aborda la implantación de una plataforma para poder operar una impresora 3D de manera remota usando una Raspberry Pi. El sistema será capaz de proporcionar datos sobre la impresión, actuar mediante comandos sobre la impresora, transmitir una señal de video en tiempo real del proceso de impresión y cargar piezas al servidor para ser impresas. Además el sistema ha de funcionar con cualquier impresora doméstica, eliminando problemas de compatibilidad en el caso de sustituir la impresora 3D actual.

Además, a la hora de conseguir una plataforma lo más autónoma posible, se ha desarrollado un sistema auxiliar que proporciona medidas de seguridad pasivas. En este aspecto se han instalado sensores de temperatura y luminosidad así como un detector de falta de filamento de impresión, supliendo así una de las grandes carencias de la inmensa mayoría de impresoras 3D domésticas.

Para la consecución de los objetivos arriba descritos, se han planteado una serie de hitos a lo largo del tiempo:

- Selección adecuada del *hardware* de control para el presente proyecto.
- Adaptar el hardware a las necesidades y requerimientos de la plataforma.
- Elección del software adecuado para el control remoto de la impresora 3D.
- Desarrollo de capacidades acorde a las necesidades marcadas en el proyecto.
- Proceso de documentación detallada del proceso para permitir abrir una vía de trabajo buscando mejoras y nuevas funcionalidades.

Para la consecución de estos objetivos arriba descritos, se han definido una serie de tareas primordiales:

- Familiarización con el entorno Raspberry Pi.
- Documentación sobre proyectos de impresión 3D remota.
- Familiarización con el proceso de impresión 3D.
- Instalación del software de control de impresión 3D.
- Desarrollo del hardware con Arduino.
- Implantación en la plataforma de impresión.
- Realización de pruebas y ensayos.
- Documentación gráfica del proceso.
- Redacción de la memoria y apertura de nuevas vías de desarrollo y mejoras.

1.3 Recursos Utilizados

En esta sección se describen los principales elementos utilizados en la elaboración del proyecto:

1.3.1 Hardware

- Raspberry Pi para contener el software de impresión 3D y crear el servidor web.
- Arduino Uno para contener el software de control de las medidas de seguridad pasiva.
- BQ Witbox I para las pruebas sobre la impresora 3D.

1.3.2 Software

- Octoprint como software de impresión remota en 3D.
- Telegram Messenger para la gestión remota de la impresora vía dicha app.
- IDE Arduino.
- Siemens NX para el diseño de ciertos elementos.

1.4 Estructura del proyecto

En este apartado se pretende presentar de manera básica la plataforma desarrollada para la impresión 3D de manera remota, mostrando un diagrama de la topología de la plataforma así como una breve descripción del sistema implementado sobre la impresora 3D

En lo concerniente a la topología, se ha optado por desarrollar dos sistemas, que si bien pueden funcionar en paralelo, se complementan para crear un sistema completo y cohesionado. Estos sistemas son la plataforma que permite operar la impresora 3D mediante un proceso remoto y el sistema auxiliar que monitoriza los factores externos a la impresión 3D pero que tienen una influencia sobre el proceso. Estos sistemas generan procesos que modifican a la impresora, tanto en el proceso de impresión como

en los procesos externos; de esta manera se consigue un sistema con un alto grado de autonomía. A continuación se muestra un diagrama que muestra de manera muy básica el funcionamiento de la plataforma al completo: (Figura 1-3)

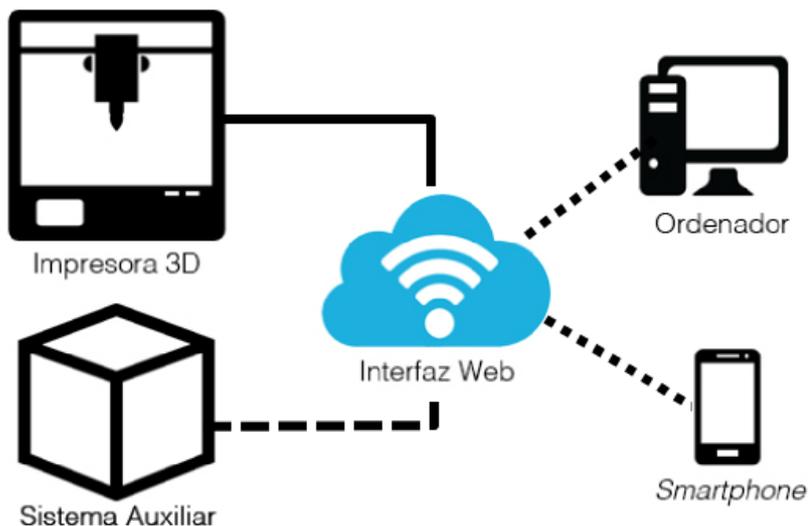


Figura 1-3 Topología básica de la plataforma de impresión 3D remota

1.5 Estructura de la memoria

El documento se ha dividido en cinco secciones:

La primera sección proporciona una visión general de la situación actual de la impresión 3D, en concreto el control remoto de la misma, así como la motivación del presente proyecto. Además, se incluyen los objetivos del proyecto así como una lista detallada de tareas que se han establecido como principales a la hora de la consecución del trabajo.

La segunda sección incluye el estado del arte de la impresión 3D. Desde la historia hasta las actuales capacidades, pasando por los distintos tipos y tecnologías relacionadas. Además se explora en la historia del hardware que principalmente se ha utilizado, Raspberry Pi y Arduino. También se proporcionan sus capacidades y su relación actual con el mundo de la impresión 3D.

En la tercera sección, el cuerpo principal del proyecto, se incluye el desarrollo de la plataforma de impresión remota. Se divide en dos grandes apartados, el *software* de impresión remota basado en Raspberry Pi y el sistema auxiliar de impresión, basado en Arduino. Se aborda la instalación y la configuración ideal del sistema. Además se incluye la introducción de mejoras en el sistema para incrementar sus capacidades iniciales.

La cuarta sección incluye las pruebas realizadas con el software de impresión 3D remota y el sistema auxiliar, una descripción de las dificultades encontradas así como de las debilidades que se han advertido en el sistema. Por último, se realiza una revisión comparativa entre el servicio de impresión remota propuesto y el servicio clásico habitual.

En la quinta sección, la última de esta memoria, en la que se recapitula a modo de resumen, se sacan conclusiones del sistema desarrollado y se abren líneas futuras para continuar mejorando el sistema en aras de conseguir una mayor automatización del sistema.

2 ESTADO DEL ARTE

En este apartado se hará una revisión de lo que ha supuesto, supone y supondrá la impresión 3D tanto de manera industrial como doméstica. Se hará especial hincapié en la capacidad de impresión 3D de manera remota y se expondrán las diferentes soluciones posibles que existen actualmente; así mismo se determinará cual supone la mejor opción.

2.1 Historia de la impresión 3D

2.1.1 Introducción

La impresión 3D más primitiva se remonta a 1976, año en el que sale al mercado la primera impresora de inyección de material. A partir de ese momento permanece patente la idea de la impresión de objetos mediante diferentes tecnologías. Paulatinamente, la tecnología avanza para pasar de la inyección con tinta a la inyección de materiales a través de diversas técnicas, hasta llegar a darle actualmente usos en campos tan dispares como la medicina o la construcción.

2.1.2 Evolución de las impresoras 3D

En 1984, Charles Hull inventa la estereolitografía (SLA) y no es hasta 1984 cuando se lanza al mercado la primera impresora 3D del tipo SLA desarrollada por la empresa *3D Systems*. Más adelante, en 1989, Scott Crump, dueño de *Stratasys*, inventa y patenta la tecnología de impresión por deposición fundida o FDM, método que más adelante democratizaría y acercaría al hogar la impresión 3D.

Ya en el año 2006 se construye la primera máquina de sintetización láser selectivo o SLS a nivel comercial; es este modelo el que abre las puertas a la industria a la personalización masiva y a la demanda de fabricación de piezas industriales. Se abre el abanico de materiales posibles y se comienzan a imprimir prótesis médicas.

Es necesario mencionar a *Industrias MakerBot*, una compañía de hardware de código abierto que en el año 2009, comienza a vender kits de montaje de impresoras 3D domésticas como el de la Figura 2-1, basadas en la tecnología FDM comenzando de esta manera la fiebre del movimiento "*Do It Yourself*". [3]

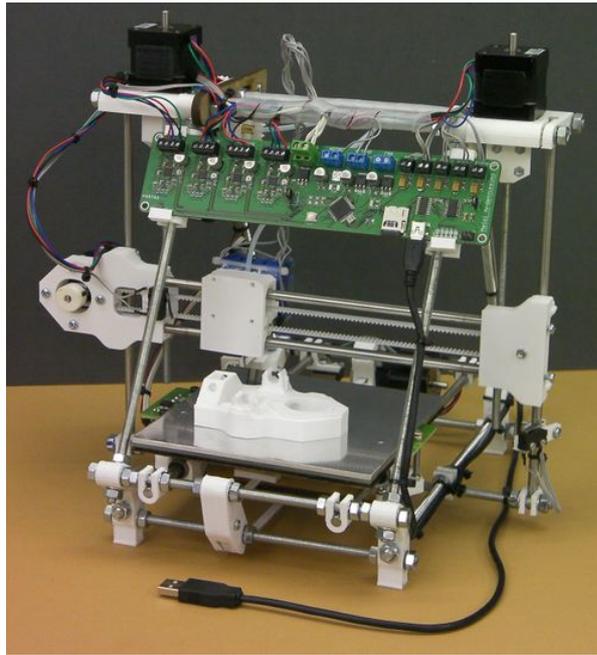


Figura 2-1 Primer prototipo de la impresora FDM "ReRap" [4]

Es en este mismo año, cuando crece de manera significativa la bio-impresión, se utilizan impresoras para crear vasos sanguíneos y órganos funcionales. Ya en 2014 y tras más de 30.000 horas en el banco de pruebas, se lanza al espacio la *Zero Gravity Printer* [5], la primera impresora 3D en el espacio. A día de hoy, las impresoras 3D están llamadas a cambiar la Industria, la fabricación aditiva es cada vez más común y a medida que el precio de adquisición baja, su implantación está asegurada.

2.1.3 Impresoras 3D industriales

La impresión 3D a nivel industrial es sin duda un mercado en expansión, en poco tiempo esta tecnología ha traspasado los departamentos de I+D+i para implantarse como una alternativa real en las líneas de producción. La creciente cantidad de materiales usados en la impresión, donde ya no se limita a polímeros sino que también puede utilizar ciertas aleaciones, la resolución cada vez mejor que permite reproducir de manera muy fiel superficies parametrizadas y el reducido coste de producción convierten a la impresión 3D en proceso productivo a tener en cuenta.

Algunas de las industrias que ya han implantado esta tecnología son por ejemplo la aeronáutica, donde Airbus en el año 2015 ya imprimió más de mil piezas del nuevo avión comercial A350 (Figura 2-2); el responsable de producción, Eric Beldin menciona sobre la impresión 3D: “Puede fabricar capa a capa, en lugar de fabricar a partir de un bloque de material, puede fabricar el interior con una estructura que permitirá que sea mucho más ligero. Por lo tanto, la ventaja para el fabricante será un consumo de combustible más reducido”. [6]



Figura 2-2 Boeing A350 [7]

Otras industrias como la automoción, la arquitectura o la medicina (Figura 2-3) tampoco se quedan atrás en este aspecto, la impresión 3D facilita y acelera en gran medida el proceso de prototipado creando así un nuevo paradigma en este ámbito. [8]

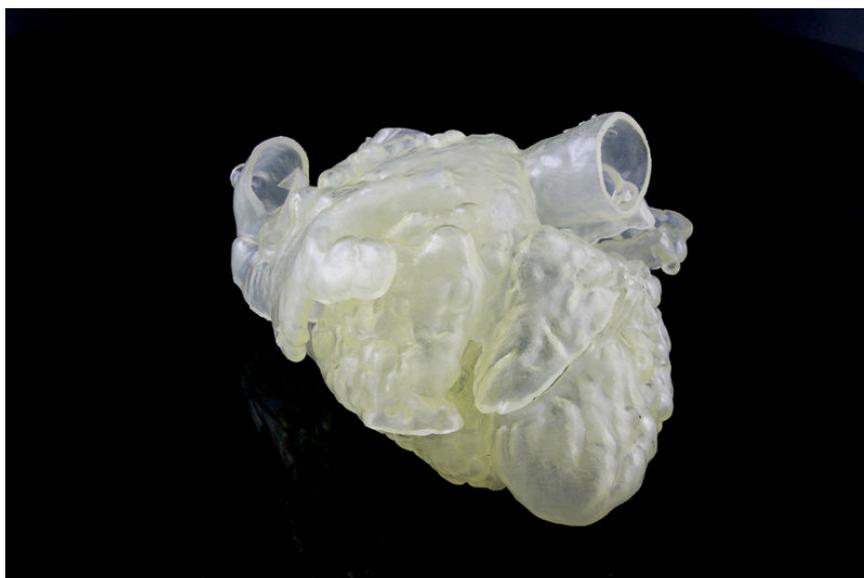


Figura 2-3 Corazón humano impreso en 3D [6]

2.1.4 Técnicas de impresión 3D en la Industria

Algunas de las principales técnicas de impresión utilizadas a nivel industrial son:

2.1.4.1 Estereolitografía (SLA)

La estereolitografía es un proceso de fabricación capa a capa inventado en 1986. Desde entonces ha sido utilizado de manera regular en distintas industrias. El proceso de fabricación se basa en un haz de luz ultravioleta que se focaliza sobre una cuba de líquido foto-polimérico, estos rayos van "dibujando" capa a capa la superficie de la pieza, el haz de luz UV es el encargado de sintetizar el líquido creando un modelo sólido. De esta manera la capa recién solidificada queda adherida a la inmediata anterior debajo de ella. En la Figura 2-4, se puede observar el esquema de funcionamiento.

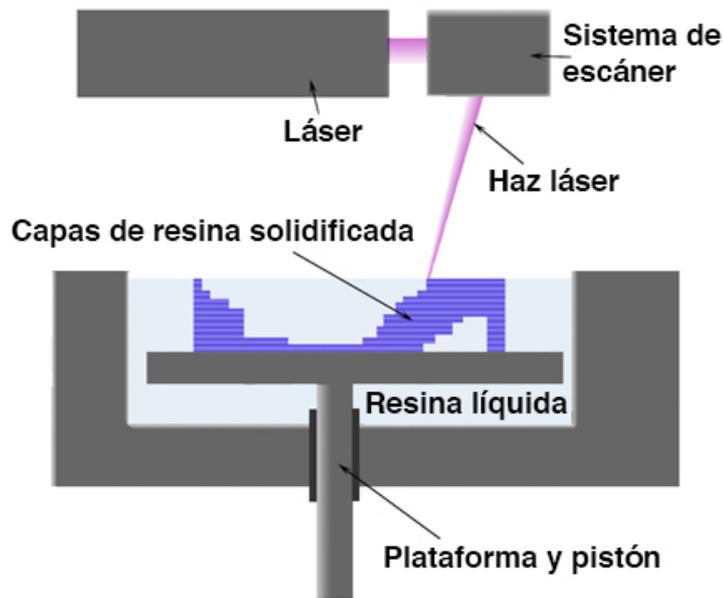


Figura 2-4 Esquema del proceso de impresión SLA [9]

La duración del proceso depende de la pieza a producir, puede variar entre unas horas y un día y los tamaños máximos rondan los 210×70×80 cm. En lo referente a la calidad de las piezas creadas, se consiguen productos con buen acabado superficial, de superficie no porosa, lo cual es una ventaja respecto a otras técnicas de impresión 3D y con una dureza tal que son aptas para ser mecanizadas, así como también para crear moldes maestros para termo-conformado, soplado o inyección.

Una de las principales desventajas de esta técnica es que las resinas utilizadas son de dureza variable, al estar expuestas a la luz ultravioleta ambiental, las piezas van volviéndose cada vez más frágiles. [10]

2.1.4.2 Sinterización selectiva por láser (SLS)

La sinterización selectiva por láser fue inventada a mediados de los años 80 por Joe Beaman en colaboración con la Agencia DARPA. La motivación del proyecto fue la de aportar flexibilidad al proceso constructivo de piezas principalmente para el entorno de Defensa.

En cuanto al proceso, se requiere un láser de alta potencia (de CO_2 normalmente) para producir la fusión de pequeñas partículas de plástico, cerámica, cristal o diversos metales. El láser genera capa a capa secciones transversales de la pieza a construir. Una vez la capa es sólida, todo el conjunto desciende una distancia igual al espesor de la capa producida; llegados a este punto, se añade una nueva capa de material y se solidifica. Este proceso se repite tantas veces como sea necesario para construir la pieza. En la Figura 2-5, se puede observar el esquema de funcionamiento.

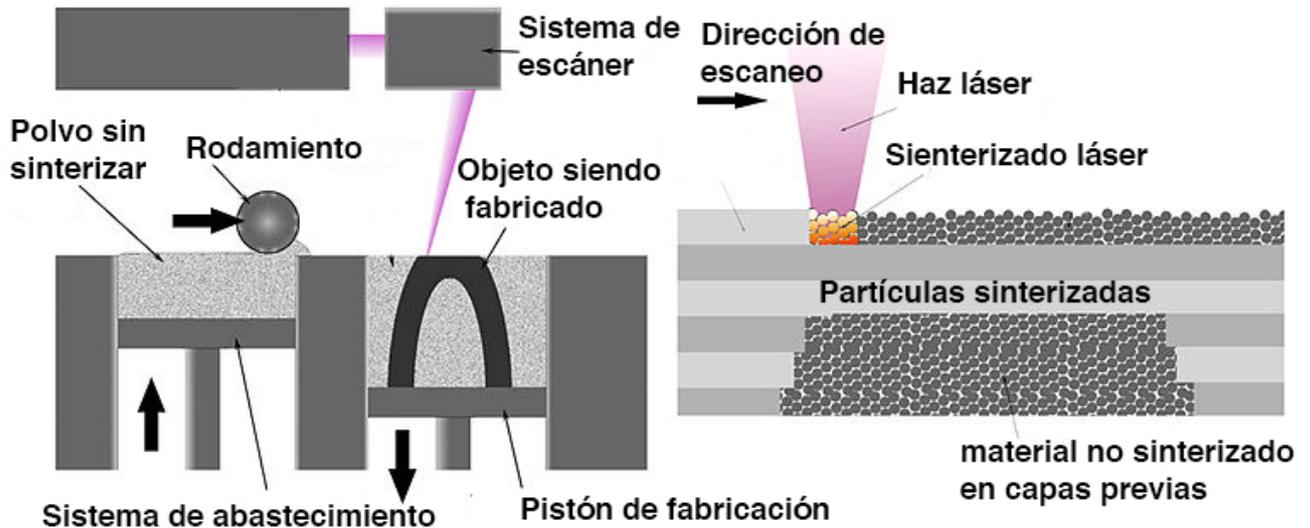


Figura 2-5 Esquema del proceso de impresión SLS [11]

Comparado con otros procesos de fabricación aditiva, el SLS es sin duda el que más posibilidades ofrece a la hora de producir piezas con un número relativamente alto de materiales en polvo a elegir. Dependiendo del proceso y del material, se pueden llegar a conseguir densidades del 100% de la densidad del material como tal en su forma sólida. Esto le confiere unas propiedades físicas comparables a aquellas fabricadas por CNC o métodos tradicionales.

La tecnología SLS se está expandiendo rápidamente debido a las posibilidades que ofrece para generar modelos complejos a partir de los modelos CAD. Mientras que en el pasado solo era utilizada para prototipado rápido de piezas, cada vez más se utiliza para tiradas cortas de piezas para uso comercial. [11]

2.1.5 Impresoras 3D domésticas

Para comenzar a hablar de la implantación en el hogar de la impresión 3D hay que remontarse al año 2000, momento en el que surge el término *fabbers*, contracción que significa "*fabricante digital*"; mediante este término se hace referencia a aquellas máquinas capaces de convertir datos digitales en productos tangibles. En ese año las impresoras 3D oscilaban entre 50.000 y 800.000 euros, algo que hizo que a nivel doméstico esta tecnología apenas creciera.

Ya en el año 2006, los avances tecnológicos permiten llevar los modelos producción industrial a la manufactura personal, nace la cultura *maker*, el ya mencionado "*Do It Yourself*", un movimiento social en el que sus adeptos crean y recrean productos sin incurrir en grandes inversiones; la cultura de la innovación y el conocimiento abierto. [12]

Aprovechando estos avances nace *Makerbot* [4] en el verano de 2012, la primera empresa de hardware abierto que penetra de manera contundente en el mercado de las impresoras 3D de bajo coste. A día de hoy existen más de 130 empresas que comercializan kits para construir una impresora 3D y se puede conseguir una por poco más de 100 euros, una auténtica revolución en solo una década. En la Figura 2-6 Principales empresas de fabricación de impresoras 3D se muestran las principales empresas que comercializan impresoras 3D.

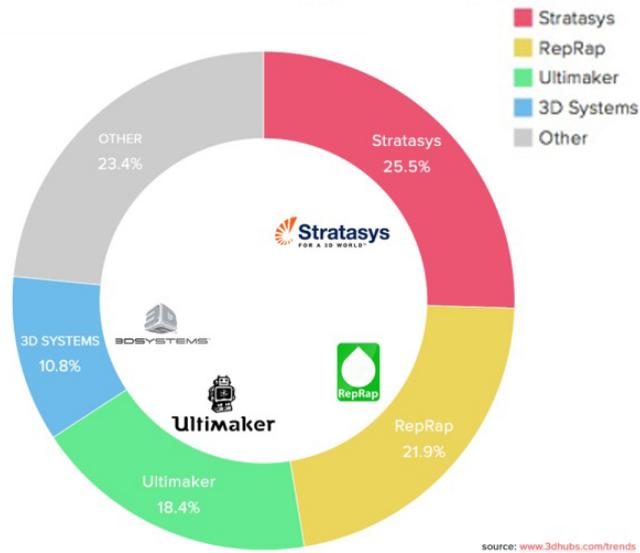


Figura 2-6 Principales empresas de fabricación de impresoras 3D [13]

2.1.6 Técnicas de impresión 3D en el entorno doméstico

El proceso de fabricación que se va a tratar en este apartado es el FDM o modelado por deposición fundida, el tipo de conformado que utilizan la gran mayoría de impresoras 3D en el hogar o en entornos no profesionales.

2.1.6.1 Modelado por Deposición Fundida (FDM)

La tecnología de modelado por deposición fundida se inventa en la década de los 80 y no se comercializa hasta 1990, en cambio no será hasta los comienzos del siglo XXI cuando realmente se generalice su uso a todos los niveles.

El proceso del FDM parte de un fichero CAD con extensión *.stl* que es orientado para ser impreso, dividiéndolo en capas y calculando las trayectorias que seguirá el extrusor. La boquilla del extrusor será la encargada de fundir el material de impresión e ir depositando el material capa a capa para construir la pieza final. La resolución máxima de la impresora es un parámetro de diseño y esta relacionada con el espesor de cada una de las capas. [14]

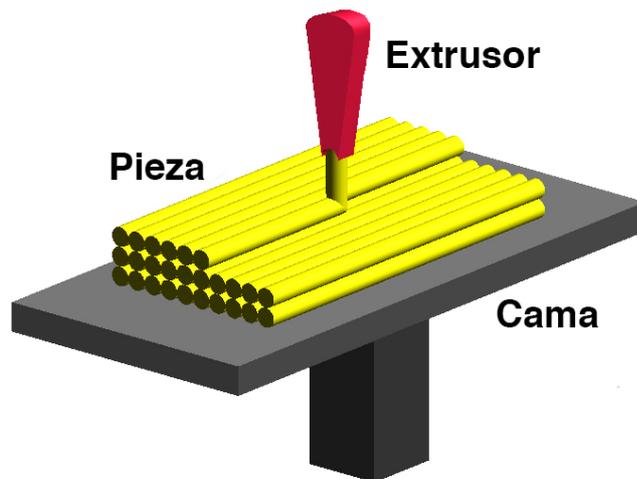


Figura 2-7 Esquema del proceso de impresión FDM [14]

A pesar de que esta tecnología resulta muy flexible, existen restricciones constructivas en las dimensiones, debido al tamaño limitado de la superficie de impresión, y en la forma, debido a las limitaciones que ofrece el polímero en lo relativo al tiempo de solidificación del mismo.

Los tiempos de impresión varían mucho con las dimensiones y forma de la pieza, puede variar desde unos minutos hasta varias horas y lo más materiales más utilizados son polímeros como el Acrilonitrilo butadieno estireno (ABS) o el Poliacido Láctico (PLA), sin embargo también se utilizan ceras chocolates (Figura 2-8) o polieterimida (PEI), un material con propiedades ignífugas.

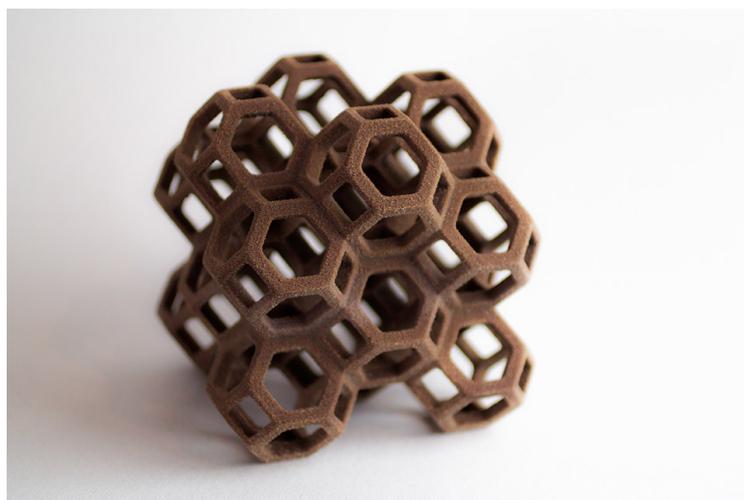


Figura 2-8 Figura de chocolate impresa en 3D [15]

Merece la pena dedicar un párrafo a unas "impresoras 3D" muy particulares, los "3D Doodler" un "bolígrafo" (Figura 2-9) que mediante una resistencia, funde filamento polimérico de ABS y permite "dibujar" utilizando las tres dimensiones.

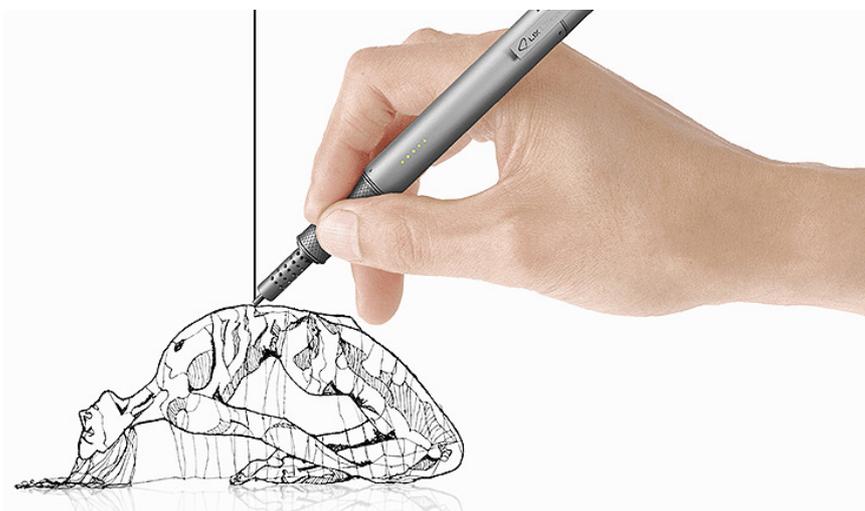


Figura 2-9 Escultura tridimensional realizada con un "bolígrafo 3D" [16]

Su creador fue la empresa WobbleWorks en 2012 a través de una campaña "KickStarter", una página web donde cualquier persona puede apoyar un proyecto que considera interesante. Se mantiene durante 3 años en desarrollo y en enero del 2015, la campaña consigue recaudar más de 1,5 millones de dólares y el "3D Doodler" se lanza de manera definitiva al mercado. [16]

En el presente existen decenas de dispositivos que permiten la impresión 3D a partir de un "bolígrafo". En cuanto a su uso, al margen del valor artístico que puede tener, supone una manera muy rápida de realizar prototipos en tres dimensiones sin la necesidad de partir de un modelo CAD, en

campos como la arquitectura o la ingeniería de diseño puede suponer un valor añadido a la etapa de diseño de cualquier proyecto. [17]

2.2 Estado Actual de la impresión 3D

2.2.1 Introducción

Resulta evidente que la impresión 3D vive una constante evolución; las nuevas aplicaciones en lo respectivo a nuevas formas y materiales o la gran tarea de marketing que están realizando las grandes industrias, posicionan a la tecnología de impresión 3D en un puesto preferente en cuanto a capacidad de evolución y desarrollo.

2.2.2 Efectos sobre la estructura de producción

No está claro si la impresión 3D está sufriendo una evolución o una revolución. Sin embargo, es seguro que va su desarrollo va a seguir avanzando y con ella los modelos de negocio convencionales, los cuales se van a ver modificados debido a los nuevas bases que está sentando esta tecnología.

En primer lugar, cabe mencionar que el segmento en el que más está creciendo la fabricación aditiva a nivel industrial es en la fabricación de productos finales, ya que en la fase de prototipado ya se puede considerar consolidada. A medida que los costes bajan y las capacidades aumentan, el abanico de componentes susceptibles de ser producido por fabricación aditiva crece de manera exponencial, especialmente aquellos con mano de obra costosa y con requisitos complejos.

Consecuentemente, la personalización masiva va a cambiar las reglas de muchos negocios; las cadenas de valor pasan a ser mucho más elásticas llevando la fabricación automatizada más cerca del consumidor, una medida que cambiaría radicalmente el modelo de industria global al "volver" el proceso de fabricación a los países desarrollados desde donde se deslocalizó en el pasado. Esto provocaría menores requerimientos de transporte del producto final cambiando totalmente el escenario geopolítico. Además, el auge de los bienes inmateriales (*crowdfunding*, *peer-to-peer*, *opensource*) tendrá repercusiones para un modelo de industria basado típicamente en el mundo físico. Según un artículo de *Le Monde* (Vaudano, 2013), los fabricantes de juguetes *Legó* y *Playmobil* mostraban preocupación por la vulneración de los derechos de propiedad industrial que significaría la impresión en el domicilio de sus juguetes (Figura 2-10).

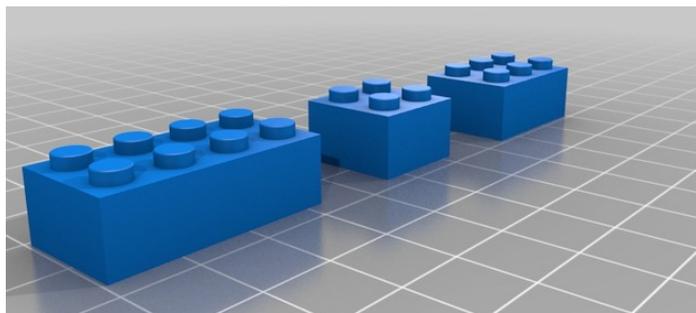


Figura 2-10 Piezas de Lego mediante diseño CAD para su impresión [1]

A este propósito se puede extraer que la tendencia natural debe ser la de concentrar el valor en el diseño principalmente y adaptar las fases de ingeniería y desarrollo a los nuevos estándares. Así como el desarrollo tradicional se basa en la estandarización, la impresión 3D facilita la personalización y elimina el concepto de producción en masa para cierto tipo de productos; el valor añadido que aporta esta tecnología puede ser el de adaptar la fabricación a las necesidades personales y demográficas, incluyendo al consumidor en el proceso de diseño y manufactura.

Un estudio realizado para IBM (Brody et al. 2013) afirma que mientras que en el actualidad los vectores de negocio de impresión 3D industrial y doméstica hay que considerarlos de forma separada, convergerán en el futuro. Además enmarca la impresión 3D en la que denomina una nueva cadena de

suministro digital (*new software defined-supply chain*) y habla de una evolución industrial a través de tres revoluciones tecnológicas:

- 3D: Los nuevos esquemas basados en la personalización eliminan el concepto de producción en masa llevando la producción hacia el terreno del *Just in Time*, la fabricación flexible y la calidad *absoluta*.
- Robótica inteligente: Esta tecnología flexible dotará a la industria de sencillez en la cadena de producción, robots que se mueven a lo largo de la cadena de montaje y no al contrario. Por lo tanto ya no será necesario que el proveedor aporte módulos complejos.
- *Open source electronics*: El *software* libre acelera la penetración en el mercado y la digitalización puesto que permite *hardware* personalizado y una mayor eficiencia al hacer accesible dicho *software* a todo el mundo. La innovación es más rápida y con aportación directa del consumidor

Sobre 2022, estas tres tecnologías permitirán una reducción del 23% (en comparación con 2012) en el coste unitario de una lavadora y en torno a un 90% en el volumen mínimo de producción de dicho electrodoméstico. Todos estos factores anteriormente descritos también tienen un efecto directo sobre la sostenibilidad medioambiental. El aprovechamiento de los materiales puede propiciar una economía circular, es decir, aquella basada en el aprovechamiento de los recursos materiales, energéticos y laborales, además la eliminación de las grandes series de producción puede aportar una producción consumiendo solamente los recursos estrictamente necesarios. [18]

2.2.3 Impresión 3D aplicada al ámbito militar

Al igual que en el ámbito civil, la fabricación aditiva está teniendo gran adopción en el ámbito militar. El desarrollo de tecnologías de fabricación de bajo coste tales como la impresión 3D, están cambiando de manera dramática el carácter del conflicto bélico tal y como lo conocemos. Según el artículo [19], el acceso por parte de entidades políticas menores a armas pequeñas, baratas e "inteligentes" tiene implicaciones en la manera de afrontar el conflicto por parte de las grandes potencias. A medida que la tecnología converge, el poder se diluye.

Por supuesto, cualquier Fuerza Armada debe estar muy fuertemente caracterizada por su adaptabilidad, y la impresión 3D está llamada a ser un punto de inflexión. Enjambres de drones o fabricación de piezas in-situ para realizar reparaciones (Figura 2-11) son solo algunos ejemplos de aplicaciones militares que ya están desarrollándose en los escenarios de operaciones. [20]



Figura 2-11 Pistoleta de un fusil M16 fabricado por impresión 3D [20]

2.2.3.1 Impresión 3D en las Fuerzas Armadas Españolas

A lo largo de los últimos años el Ministerio de Defensa ha comenzado a apostar por la impresión 3D y es de esperar que en años sucesivos la gestión logística de suministros y en especial, la dedicada a mantenimiento correctivo no programado, se mejore sustancialmente mediante la impresión 3D. Respecto al escenario de aplicación, caben esperar dos situaciones principalmente; por un lado, las Unidades a flote de la Armada Española, donde con mucha probabilidad en un futuro cercano se sustituirán los tradicionales tornos por impresoras 3D para la fabricación de repuestos eventuales; por otro lado, las misiones en tierra, las cuales pueden ofrecer un entorno cambiante donde una impresora 3D podría suplir las carencias de suministros. Dentro de este entorno podemos considerar el apoyo a expediciones científicas como las de la Antártida, donde en el 2016-2017 como parte del proyecto “Uso de técnicas de impresión 3D para la fabricación *in situ* de repuestos” financiado por el Centro Universitario de la Defensa de Zaragoza (CUD, 2013-2017) se ha incorporado a la expedición una impresora 3D del tipo FDM. En la Figura 2-12 se puede observar la impresora 3D utilizada en el proyecto. [21]



Figura 2-12 Impresión 3D a bordo del B.I.O Hespérides [22]

A bordo se han llevado a cabo experimentos en relación a la impresión 3D tales como la fabricación de los contenedores de los *loggers* (Figura 2-13), unos sistemas para monitorizar los movimientos que realiza el pingüino barbijo. De esta manera, se instalan réplicas del *logger* real para usar en prácticas para estudiantes y así no exponer el aparato real a posibles pérdidas o roturas durante la fase de aprendizaje. [22]



Figura 2-13 Caja de un *logger* fabricada mediante impresión 3D a bordo

Otro caso fue el de la fabricación de una herramienta concreta para un equipo de sismología. Una vez explicada la necesidad, el equipo del CUD de Zaragoza diseñó la pieza y de manera remota y casi

en tiempo real, la impresora fabricó la pieza de la Figura 2-14 sin necesidad de personal instruido en impresión 3D manipulando en el laboratorio la impresora. [22]



Figura 2-14 Reparación de un equipo con una herramienta impresa en 3D [21]

De esta manera, no cabe duda que esta tecnología evolucionará y dotará a las unidades de un medio más de gestión de suministros en campos donde hasta ahora la trazabilidad es muy pequeña. [22] [21].

2.3 Impresión 3D de manera remota

2.3.1 Introducción

Uno de los mayores inconvenientes que se pueden extraer de la impresión 3D es el elevado tiempo que por lo general le lleva a una impresora imprimir una pieza. En algunos casos puede tardar de 8 a 10 horas y lo más recomendable es tener la máquina vigilada para evitar problemas inesperados tales como cortes en el suministro eléctrico o fallos en la impresión. Para solucionar este problema, en los últimos años se han desarrollado plataformas para conseguir que en entornos domésticos en los que no puede existir personal dedicado solo a la impresora, pueda existir una vigilancia de manera remota por parte del operador.

Actualmente existen dos plataformas principales, ambas integradas en la impresora a través de una placa Raspberry Pi en su montaje más común, a continuación se va a tratar más a fondo cada uno de los sistemas.

2.3.2 Octoprint

La plataforma de código abierto Octoprint nace de la mano de Gina Häußge, una alemana cuya principal motivación fue la de adaptar las impresoras 3D al estilo de vida del siglo XXI. Con esta afirmación, Gina quiso dar a entender que las posibilidades actuales en lo relativo a la conectividad de la gran mayoría de impresoras 3D del mercado es muy limitada.

En 2013 crea una versión beta de lo que es el actual Octoprint (cuyo logo se muestra en la Figura 2-15) y muy pronto la comunidad de *makers* adopta la plataforma como una gran solución a la impresión remota en 3D. Un gran número de desarrolladores de código *Open Source* trabaja en la API y el reciente apoyo dado a la plataforma por parte de la empresa española BQ convierte a Octoprint en una solución de garantía.



Figura 2-15 Logo de Octoprint [23]

Octoprint funciona como una interfaz web (Figura 2-16) que proporciona al usuario una lectura precisa de los parámetros de impresión tales como:

- Temperaturas de la cama y del extrusor
- Visualizador de código GCODE en tiempo real
- Control de parada y marcha de la impresora
- Control de la cabeza de impresión en todos los ejes a través de la interfaz.
- Cámara web para poder observar en tiempo real el avance de la impresora.

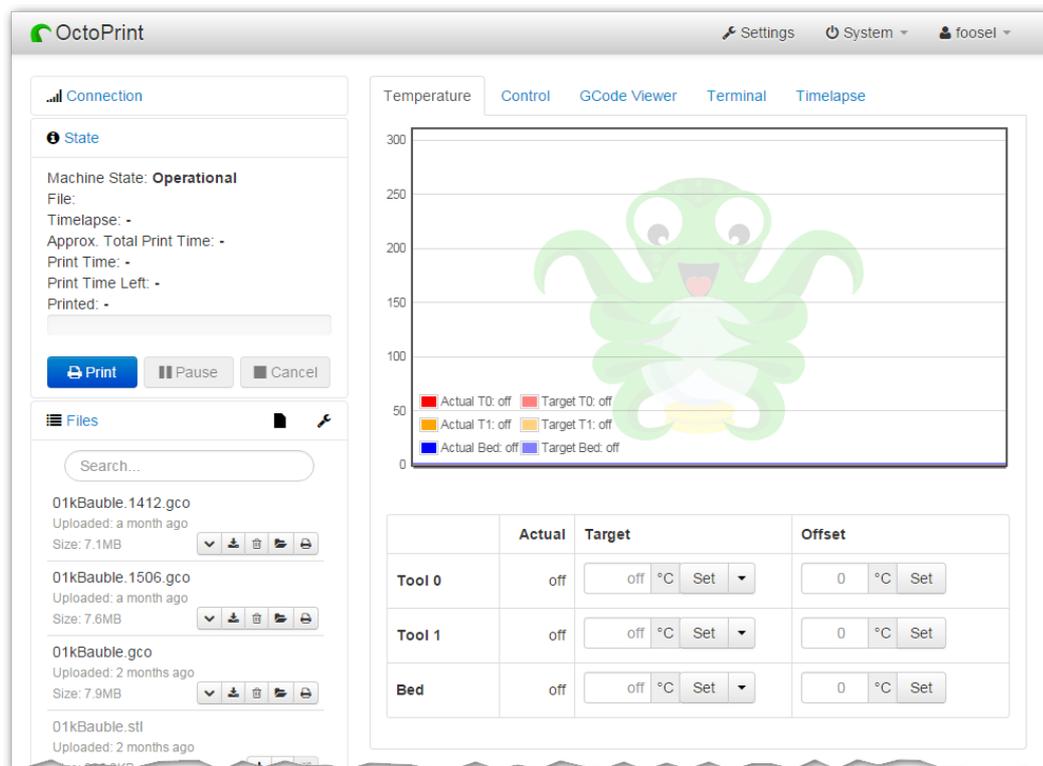


Figura 2-16 Interfaz web de Octoprint

Además, por medio de los plugins se han implementado funciones adicionales tales como las notificaciones vía *e-mail*, sms, o la capacidad de interactuar con la impresora por medio de la aplicación de mensajería Telegram.

La plataforma Octoprint se puede instalar sobre sistemas operativos Windows y Mac pero su uso más extendido es a través de la imagen "OctoPi" que contiene el sistema operativo Raspbian, una distribución de Debian optimizada para la Raspberry Pi. Este software, además del software Octoprint incluye un sistema de *streaming* de video y el *software* de impresión CuraEngine v15.04 que se ejecuta directamente en la Raspberry Pi.

De esta manera y a través de un túnel SSH entre el dispositivo sobre el que se vaya a ejecutar la interfaz web y la Raspberry Pi, que estará conectada mediante Octoprint a la impresora 3D, se produce comunicación bilateral, recibiendo notificaciones del estado de la impresora con la capacidad de ejecutar órdenes sobre esta a través de la interfaz web. Algunos de los inconvenientes más comentados de la plataforma están referidos a la complejidad en su instalación, ya que Octoprint no está diseñado para trabajar con interfaz de escritorio, sino a través de la terminal mediante comandos. Además, el proceso de conexión no es del todo intuitivo. [23]

2.3.3 Astroprint

En Mayo del 2014 y como parte de una campaña de la conocida web de desarrollo de ideas de código libre www.Kickstarter.com, nace Astroprint (cuyo logo se muestra en la Figura 2-17). La plataforma logra recaudar 40.314 dólares alcanzando su meta en un solo día. Sus creadores, Drew Taylor, CEO de la compañía, Daniel Arroyo, Joshua White y Kirsten Osolind, crean Astroprint como *fork* de Octoprint, es decir, una rama de Astroprint tomando su código fuente como base. Ambas plataformas cubren necesidades similares pero algunas funcionalidades y decisiones sobre la interfaz han variado. [24]



Figura 2-17 Logo de Astroprint [24]

En cuanto a las funcionalidades básicas del sistema, son comparables a las que ofrece la plataforma Octoprint, capacidad de operación con la impresora a distancia, monitorización de los datos de impresión vía red local o internet y recepción de imágenes en tiempo real de la impresión en curso.

En lo relativo a la interfaz, ofrece un menú más sencillo que se ha pensado para poder ser integrado y manipulado a través de dispositivos móviles. Además, algunas funcionalidades avanzadas se han omitido para facilitar la experiencia de usuario.

La conectividad también se ha resuelto de una manera más elegante en Astroprint, la conexión es más rápida y ofrece ciertas funcionalidades que no se han implementado en Octoprint pero una vez conectados, la comunicación con la Raspberry Pi es a través de un *túnel SSH* de manera idéntica al servicio Octoprint.

Una vez se comienza a imprimir es donde aparecen las diferencias más significativas, mientras que con Astropint el código *gcode* es generado por los servidores de Astroprint vía Internet, Octoprint se apoya en el *software* de impresión *CuraEngine*. Esto significa que no hay necesidad de estar conectado a internet en el proceso para el caso de Octoprint., haciendo depender al proceso de impresión de menos factores externos. [24]

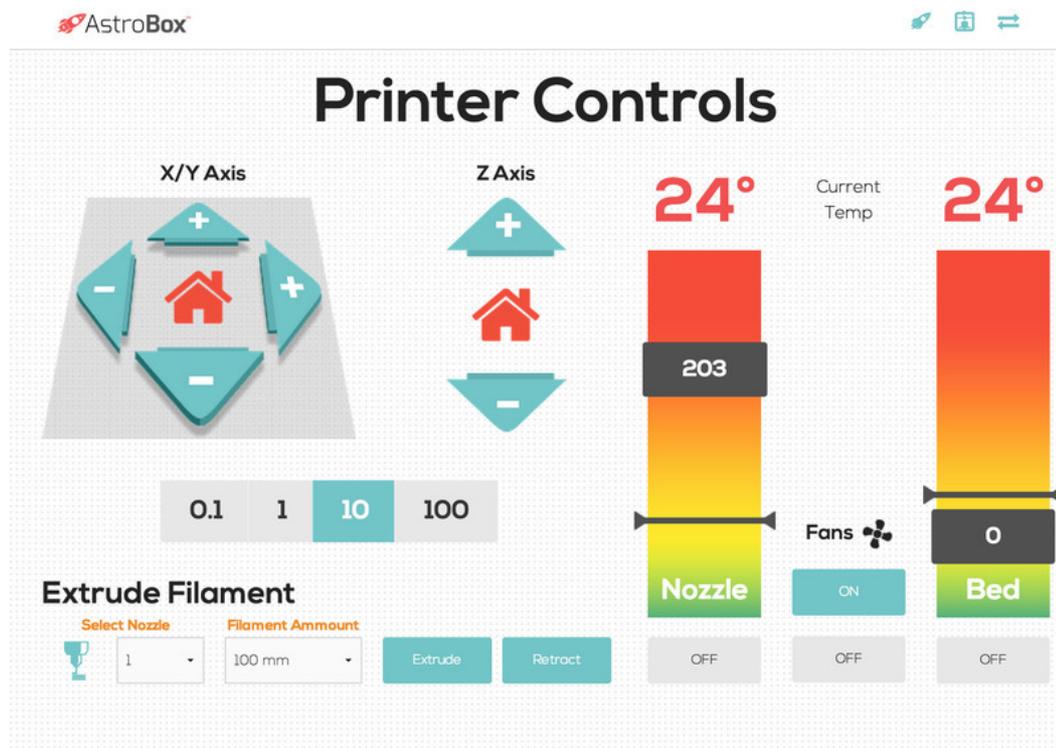


Figura 2-18 Interfaz web de Astroprint [24]

En lo relativo a los *plugins* cabe destacar que al tratarse de una plataforma más moderna, cuenta con una API menos desarrollada y a día de hoy se trata de un software poco desarrollado en este aspecto.

2.3.4 Conclusiones de los servicios de impresión de manera remota

Después de la revisión a los dos principales servicios de impresión remota, queda patente que ambos se complementan y que están hechos para dos grupos de usuarios distintos. Mientras Octoprint está ideado para el usuario que necesita funcionalidades avanzadas de su impresora para tener un control total sobre ella, Astroprint está ideado para el usuario que tiene un control medio sobre su impresora, que se centra más en tener una experiencia sencilla y rápida que un profundo conocimiento sobre los parámetros de impresión. Además, en el caso de querer operar la impresora desde el *smartphone* o la *tablet*, Astroprint es una elección más correcta.

2.4 Raspberry Pi

2.4.1 Introducción

Raspberry es el ordenador con arquitectura ARM de bajo coste más famoso del mundo. Desde su lanzamiento en el año 2012 se han vendido más de 6 millones de unidades, suponiendo de esta manera el ordenador de origen británico más vendido de todos los tiempos. Pero, ¿Qué es realmente una Raspberry Pi? Sus creadores, la *Raspberry Pi Foundation* la definen como un ordenador del tamaño de una tarjeta de crédito que puede ser usado tanto para proyectos de electrónica como para la mayoría de tareas que puede realizar un ordenador doméstico. Su logo se muestra en la Figura 2-19.

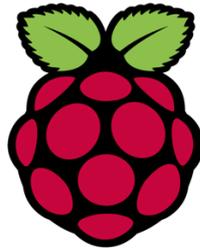


Figura 2-19 Logo de *Raspberry Pi Foundation* [25]

La *Raspberry Pi Foundation* es una organización sin ánimo de lucro que centra sus esfuerzos en facilitar la tarea del aprendizaje de lenguajes de programación a las nuevas generaciones. Su idea era la de crear la plataforma más barata posible. Otro de sus objetivos es el desarrollo del software necesario para ser un conjunto útil en la labor docente. Se ofrecen diferentes modelos de Raspberry Pi desde 5\$ hasta 35\$, lo cual supone una de las claves de su éxito. [25]

2.4.2 Historia

La idea original de la Raspberry Pi nace trabajando en el Departamento de Ciencias de la Computación de la Universidad de Cambridge, en el año 2000. Trabajando al frente del Departamento, Eben Upton advierte una tendencia negativa en los conocimientos sobre programación de los estudiantes de nuevo ingreso; Esto le motiva a fundar, junto con un grupo de con un grupo de ingenieros, informáticos y profesores, la *Raspberry Pi Foundation* en el año 2009.

Originalmente se planea lanzar al mercado dos modelos, el *Model A* y el *Model B* mostrados en la Figura 2-20, por diversos inconvenientes relacionados con las licencias y la fabricación, solo es presentado el *Model B* en un principio. La diferencia entre estos dos modelos radica en el hardware y en el precio, mientras que el *Model A* costaba 25\$, el *Model B* incluía un puerto Ethernet y un puerto USB extra por 35\$. [25]

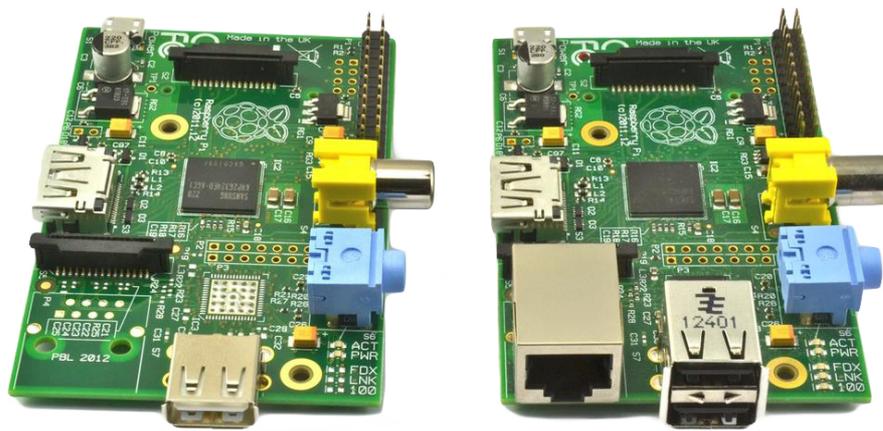


Figura 2-20 Comparación de la Raspberry Pi Model A y model B [25]

Desde el principio la plataforma es un rotundo éxito. La Fundación vendió todas las unidades inicialmente fabricadas por un precio ligeramente superior al de coste, debido al origen no lucrativo de la Fundación. Sin embargo, el propio Upton explicaba que los beneficios se invierten en una gran variedad de proyectos educativos relacionados, en la mayoría de los casos, con el mundo de la informática.

La Raspberry Pi Foundation nunca dejó de innovar, tanto es así que comienzan a focalizar sus esfuerzos en una mejora significativa del software. De esta manera nace Raspbian, una distribución de Debian Linux optimizada y diseñada específicamente para la Raspberry Pi. Desde el primer

lanzamiento en 2012 se han lanzado al mercado 7 placas distintas cuyas especificaciones se detallan en la siguiente tabla: (Tabla 1)

	Raspberry Pi 1 Model A	Raspberry Pi 1 Model A+	Raspberry Pi 1 Model B	Raspberry Pi 1 Model B+	Raspberry Pi 2 Model B	Raspberry Pi 3 Model B	Raspberry Pi Zero
Puertos USB 2.0	1	1	2	4	4	4	1 (micro USB)
Ethernet	No	No	10/100 Mbit/s	10/100 Mbit/s	10/100 Mbit/s	10/100 Mbit/s	No
Bluetooth	No	No	No	No	No	4.1	No
WiFi	No	No	No	No	No	802.11n	No
Entrada de audio	I2S						
Salida de audio	jack 35mm HDMI	jack 35mm HDMI, GPIO analógico PWM					
Entrada de video	Connector de cámara CSI	No					
Salida de video	HDMI, RCA	HDMI, TRRS	Mini-HDMI, GPIO composite				
Almacenamiento Externo	Tarjeta SD	Tarjeta MicroSD	Tarjeta SD	Tarjeta MicroSD	Tarjeta MicroSD	Tarjeta MicroSD	Tarjeta MicroSD

Tabla 1 Comparativa de los modelos disponibles de Raspberry Pi

2.4.3 Estado actual y futuro de la plataforma

Dado el enorme éxito de ventas que ha supuesto la Raspberry Pi, cuenta con una numerosa comunidad detrás que la ha utilizado para proyectos y en escenarios de diversa índole, desde la creación de un centro multimedia doméstico, emuladores de videoconsolas hasta el lanzamiento de una unidad a la Estación Espacial Internacional (Figura 2-21) para el proyecto de investigación *AstroPi*. [26] [27]. Otros proyectos incluyen pequeños aviones autónomos, adaptaciones para usos en domótica e incluso un receptor FM casero. Uno de sus creadores la describe como un bloque de construcción barato para construir cualquier cosa imaginable. [27]



Figura 2-21 Módulo Astro-Pi a bordo de la Estación Espacial Internacional (EEI) [5]

Respecto al futuro de Raspberry Pi, sus creadores hablan de hacer frente a la demanda del reciente lanzamiento de Raspberry Zero (Figura 2-22) y seguir creciendo para poder ofrecer un ordenador plenamente operativo a un precio asequible para los países menos desarrollados. [25]

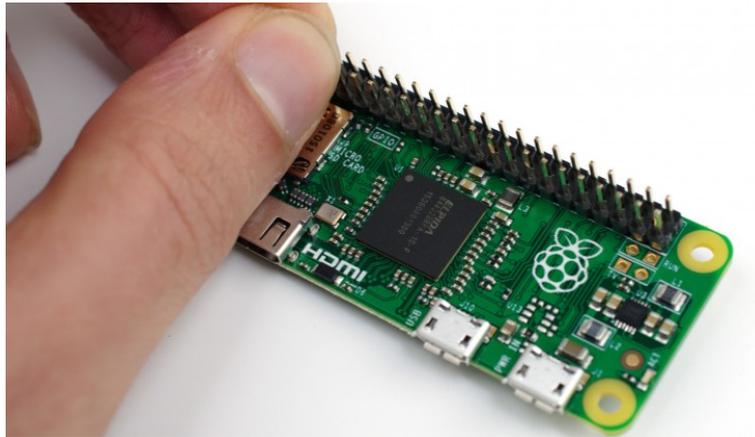


Figura 2-22 Modelo "Zero" de Raspberry Pi [25]

2.5 Impresión 3D basada en Raspberry Pi

2.5.1 Introducción

Existen multitud de proyectos de código abierto en los que la Raspberry Pi hace las veces de ordenador del sistema. Incluso algunos en los que el microcontrolador es sustituido por la propia placa Raspberry Pi para ejecutar el programa. A continuación se hará una revisión de los trabajos existentes.

2.5.2 Estado actual de los proyectos

En cuanto a la impresión 3D basada en el uso de una Raspberry Pi como controladora del sistema, se pueden hacer dos grupos principales, aquellos proyectos en los que Raspberry Pi es la generadora de la interfaz de usuario y envía los comandos *gcode*, es decir, las instrucciones de impresión necesarias para realizar la pieza. Por otro lado, las impresoras en las que Raspberry hace las veces de microcontrolador del sistema y envía a los motores y al extrusor las órdenes directamente. [23]

Respecto al primero grupo, los servicios Octoprint y Astroprint de los que ya se ha hablado ofrecen estas capacidades, actúan como interfaz entre el usuario y la impresora y ejecutan el motor de impresión que ordena a la impresora lo que debe hacer.

En lo respectivo al segundo grupo, solo existe de manera documentado un proyecto que incorpore este sistema, se trata de un proyecto de Owen Jeffreys, un estudiante de ingeniería que ha trabajado en integrar la Raspberry Pi en una impresora 3D completamente casera y de código libre. Las razones por las cuales no existen más desarrollos en este ámbito pueden ser varias, por un lado, es una tecnología aún en pleno crecimiento y nuevos proyectos surgen cada día; por otro lado, el coste que supone un microcontrolador hoy en día no es elevado, con lo cual la integración con Raspberry Pi, no es del todo fundamental. [28]

2.6 Arduino

En los siguientes apartados se va a tratar la historia de la plataforma Arduino, así como su evolución y en lo que consiste tanto el hardware, como el software que desarrollan.

2.6.1 Historia de Arduino

La Fundación Arduino es creada en el año 2005 por Maximo Banzí, David Cuartielles, Tom Igoe, Gianluca Martino, David Mellis y Nicholas Zambetti como un proyecto para estudiantes en el instituto IVREA en Ivrea, Italia. El nombre nace de un bar donde uno de sus creadores pasaba algunas horas, el logo se muestra en la Figura 2-23. La creación de la placa electrónica del mismo nombre, primer objetivo de la Fundación, es diseñada a partir de la tarjeta electrónica *Wiring*. Una vez esta plataforma dejó de desarrollarse, el equipo de Arduino tomó las riendas para hacerla más ligera, económica y disponible para toda la comunidad (*software* y *hardware* abierto). [29]

2.6.2 Descripción de la plataforma

La plataforma Arduino se refiere a tres elementos relacionados [29]:

- **Hardware:** Se han desarrollado distintas placas electrónicas bajo el nombre de Arduino, todas ellas incorporan puertos de entrada y salida y un microcontrolador. La diferencia entre todas ellas se basa a grandes rasgos en el número de puertos disponibles y en la capacidad de procesamiento y memoria *flash* de cada una de ellas. Se trata de hardware libre, esto permite el desarrollo de infinidad de proyectos
- **Entorno de programación:** En inglés conocido como IDE, (Entorno de desarrollo integrado) es una herramienta que facilita enormemente la creación de código en comparación con otros lenguajes. Está basado en la plataforma *Processing* y utiliza el lenguaje de programación *C++*. Además de ser software libre, está integrado para la gran mayoría de Sistemas Operativos como Windows, MacOS o Linux.
- **Comunidad Arduino:** Existe en internet una extensa comunidad que comparte sus proyectos y programas al alcance de todos.



Figura 2-23 Logo de *Arduino Foundation* [29]

2.6.3 Conclusión

Tras el análisis realizado, queda patente que el uso de la tecnología de código abierto está creciendo de manera exponencial en los últimos años y cuenta con una gran aceptación en la sociedad. En el caso de la impresión 3D, no se queda atrás en este aspecto; cada vez más, esta tecnología está llegando a todos los lugares y día tras día es sencillo encontrar nuevas noticias y avances a este respecto. Por último, con la llegada del *Internet de las cosas*, cada vez más, va a ser una constante la integración y la conectividad de todos los periféricos, ascendiendo, de esta manera, un escalón más en lo concerniente al desarrollo tecnológico.

3 DESARROLLO DEL TFG

En este capítulo se definirán los pasos y puntos más importantes del proceso de desarrollo de una plataforma de impresión 3D remota. Además, se describirán y configurarán los parámetros de control desde una perspectiva tanto de *hardware* como de *software*.

3.1 Descripción de la arquitectura de una plataforma remota de impresión 3D

3.1.1 Introducción y características del sistema

Basado en las plataformas actuales a disposición del usuario y de los objetivos establecidos en el apartado Objetivos, el sistema de impresión 3D remota desarrollado cuenta con las siguientes características:

- Operación de la impresora 3D de manera remota.
- Carga de archivos *.stl* a través de la red.
- Monitorización de parámetros de la impresora como temperatura de la cama o del extrusor.
- Recepción de imágenes del proceso de impresión en tiempo real.
- Movimiento del cabezal de manera manual a través de la plataforma de impresión.
- Sistema de acondicionamiento de la caja de impresión.
- Sistema automático de iluminación para correcta recepción de imágenes.
- Interfaz sencilla para que cualquier usuario autorizado pueda ser operador de la impresora.
- Protección del sistema mediante contraseña.
- Notificación de fin de impresión.

3.1.2 Diseño del sistema de impresión remota.

Existen distintas arquitecturas de conexión a la Raspberry Pi generadora de la interfaz que controla la impresora 3D, básicamente, a través de un router en red local o en internet, se le da acceso a cualquier usuario mediante un ordenador a la operación de la impresora 3D. En este caso como se muestra en la Figura 3-1, la Raspberry Pi mediante conexión física a la impresora y a la cámara web, ejecuta el software de impresión remota. Por otro lado un ordenador conectado a la misma red que la Raspberry Pi, puede acceder a la interfaz web de impresión mediante un dominio con extensión *.local*.

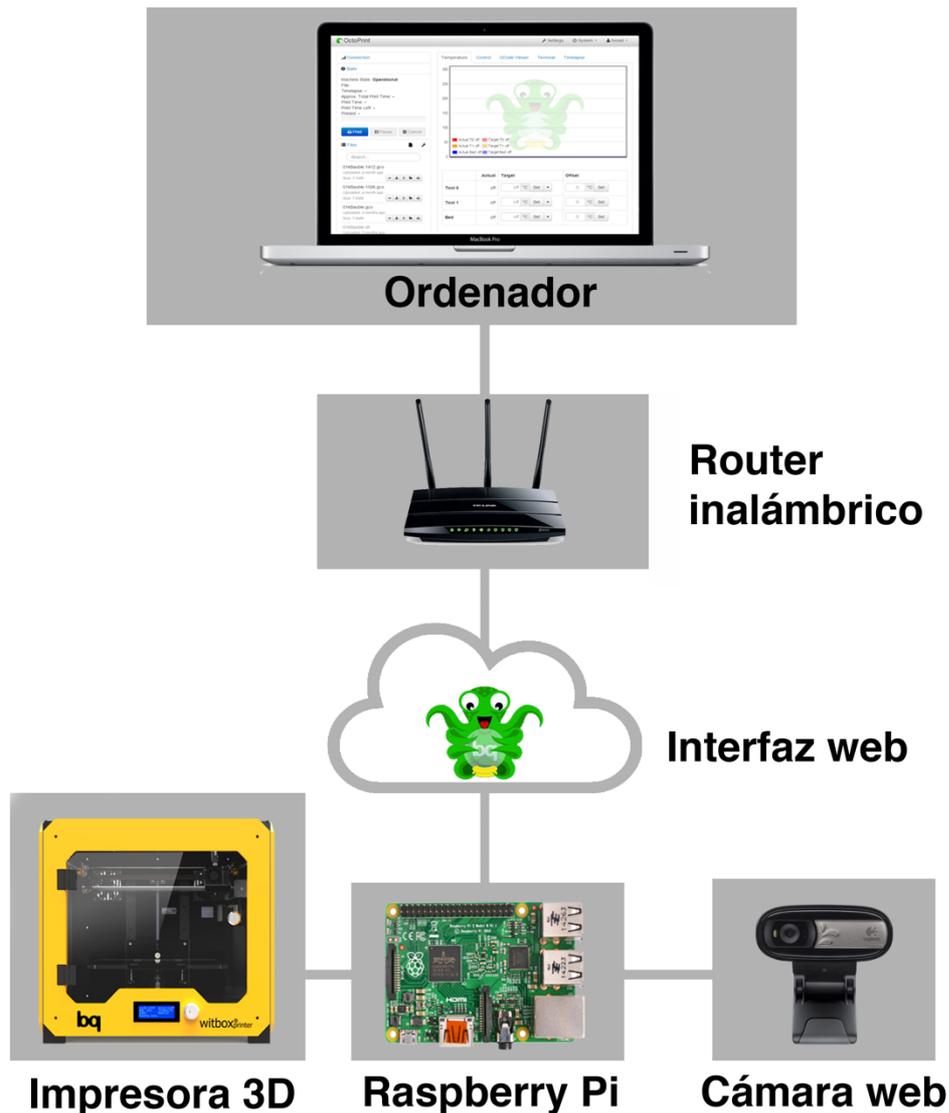


Figura 3-1 Topología de la plataforma de impresión remota en 3D

Existen diversas particularidades acerca de la topología mostrada en la Figura 3-1 que merece la pena tomar en consideración para asegurar el buen funcionamiento del sistema:

- Debido a las limitaciones gráficas de la Raspberry Pi, es recomendable para ejecutar el software de impresión remota con fiabilidad, contar con los modelos 2 o 3 de la misma.
- Aunque la cámara web puede transmitir a 24 fotogramas por segundo, con una lectura mucho menor (4 fps), se pueden conseguir los resultados deseados, una confirmación visual del resultado de la impresión a cada instante.
- La interfaz web puede ser ejecutada a través de más de un ordenador al mismo tiempo, resulta conveniente no hacerlo para evitar errores cruzados.

3.1.3 Descripción del sistema de impresión 3D remota

Como elementos en el desarrollo del sistema de impresión 3D remota se han utilizado los siguientes componentes:

Raspberry Pi:

El elemento central del sistema es una *Raspberry Pi model B*. Utiliza el Sistema Operativo Raspbian, una variante de Debian optimizado para Raspberry Pi, basado en Linux conocido por su

estabilidad y escalabilidad. Aunque utiliza Raspbian como S.O., la instalación se lleva a cabo mediante la imagen "OctoPi", la cual lleva asentada sobre Raspbian los sistemas *Octoprint*, un sistema de *streaming* de video MPEG y el software de impresión 3D *CuraEngine 15.04*. Los detalles de la instalación de *OctoPi* se explican más adelante, en el apartado 3.4.1.

Impresora 3D: Todas las pruebas se han realizado con una *BQ Witbox I* (Figura 3-2) como impresora 3D del sistema. Sin embargo, prácticamente todas las impresoras 3D domésticas soportan el software de impresión remota *Octoprint*, algo que simplifica la integración del software en el caso de cambio de impresora. Las especificaciones técnicas de la impresora BQ Witbox I se anexan en el apartado 0.



Figura 3-2 Impresora 3D Witbox I [30]

Cámara web: La cámara web *Logitech C170* (Figura 3-3) se ha utilizado para obtener señal de video del proceso de impresión. Sus características se especifican en el apartado 0.



Figura 3-3 Cámara web Logitech C170

Router inalámbrico: El *router* inalámbrico 2,4 GHz TP LINK MR200 que como se puede observar en Figura 3-1, comunican a la Raspberry Pi y al ordenador mediante una dirección IP estática proporcionada por el mismo. Además, proporciona salida a internet a la Raspberry Pi, para poder utilizar la interfaz web fuera de la propia red local. Sus características se anexan en el apartado 0.



Figura 3-4 Montaje del router en la plataforma de impresión remota

Ordenador: El ordenador utilizado sirve como manipulador de la interfaz web de impresión 3D. En el caso de impresión remota en red local, está conectado mediante Wi-Fi o el puerto Ethernet al router inalámbrico. En la Figura 3-5 se muestra el montaje con el ordenador a la hora de realizar las pruebas.

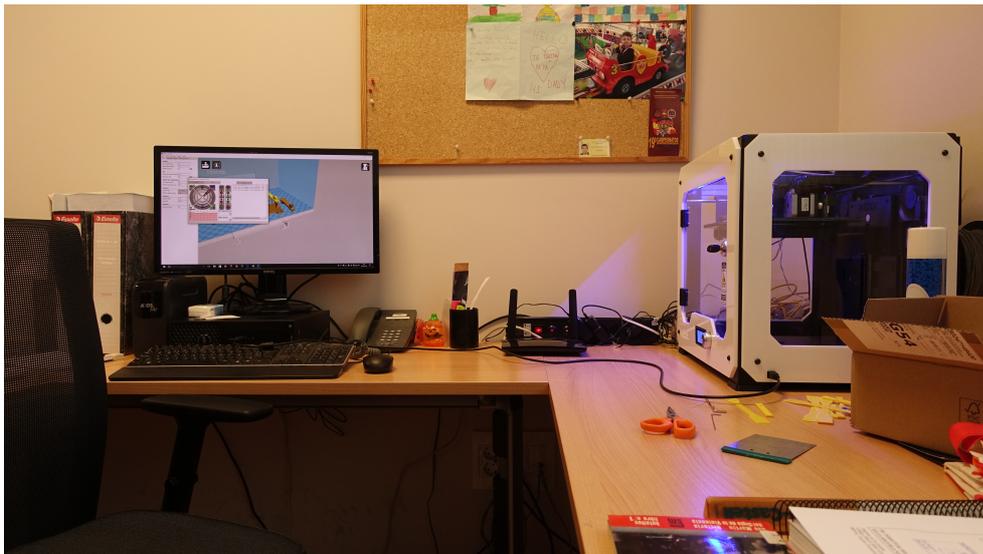


Figura 3-5 Montaje del sistema en la impresora 3D

En cuanto a los elementos utilizados en los sistemas de seguridad pasiva implementados en la impresora se han utilizado los siguientes componentes:

3.1.4 Descripción de la interfaz de red del sistema de impresión 3D remota

En cuanto a las conexiones del sistema, es conveniente establecer dos grandes grupos, la red LAN o de área local, que va a ser la red habitual con la que trabaje el sistema de impresión, y la red WAN, es decir, el acceso a través de internet al servicio de impresión. En lo relativo a esta última, es conveniente tomar ciertas seguridades ya que el control total de una impresora 3D de manera relativamente accesible en la red, ofrece ciertas vulnerabilidades. Las conexiones se han realizado a través de un router inalámbrico como se describe a continuación en la Figura 3-6.

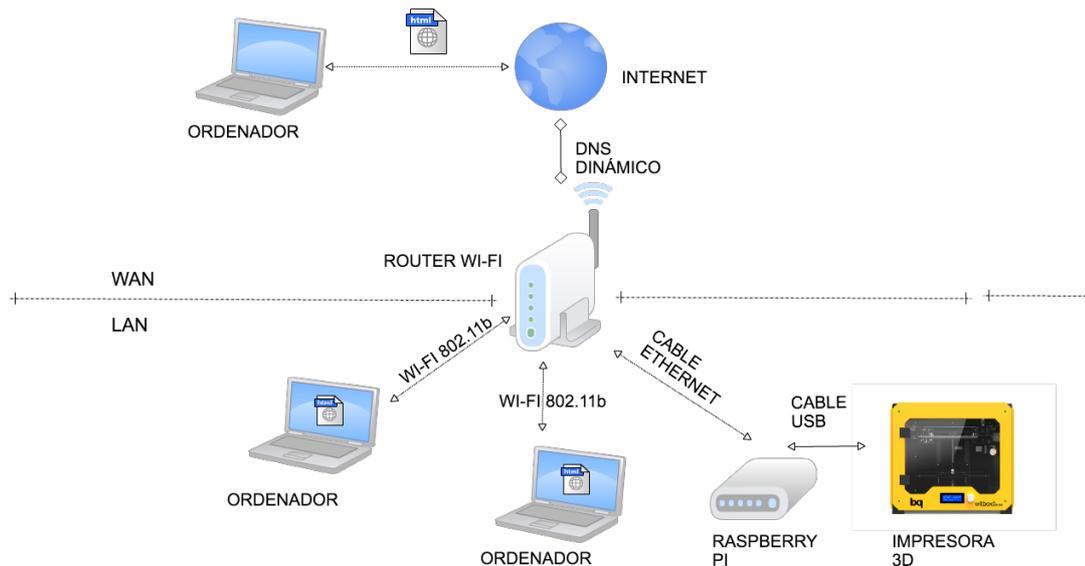


Figura 3-6 Descripción de la interfaz de red del sistema de impresión remota

3.1.4.1 Red LAN (Local Area Network) del sistema de impresión

Como se puede observar en la Figura 3-6, un *router* inalámbrico es el encargado de crear una red wi-fi que permite el acceso a la plataforma web crea Raspberry Pi a los dispositivos conectados a esa red. Respecto a la conexión de la Raspberry Pi a la red LAN, se ha optado por una conexión alámbrica mediante un cable Ethernet al *router* para aportar robustez al sistema.

La impresora 3D está conectada a la placa Raspberry Pi, controladora del sistema de impresión remota, mediante un cable USB que permite la comunicación bilateral Raspberry Pi-impresora 3D.

Es conveniente mencionar que para evitar ciertos conflictos de red, es recomendable que solo un dispositivo acceda al servicio de impresión a la vez; para realizar esta comprobación bastará con ver el apartado **status** de la interfaz web como se indica en el apartado 3.3 una vez se accede a la interfaz web Octoprint. El acceso a la interfaz web se realiza mediante un navegador web del ordenador conectado a la red LAN. Para el proceso de resolución de la dirección se ofrecen dos posibilidades:

- Introducción en el navegador de la dirección IP de la Raspberry Pi, la cual se puede conocer introduciendo en el *Shell* de Raspbian el comando : `ifconfig` (Figura 3-7)

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Oct 25 14:38:05 2013 from davids-iphone
pi@raspberrypi ~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:f2:03:a1
          inet addr:192.168.1.127  cast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:17383 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9236 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1151561 (1.0 MiB)  TX bytes:2844379 (2.7 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:43341 errors:0 dropped:0 overruns:0 frame:0
          TX packets:43341 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:109181135 (104.1 MiB)  TX bytes:109181135 (104.1 MiB)
```

Figura 3-7 Respuesta al comando `ifconfig` en la *shell* de Raspbian

En el ejemplo de la Figura 3-7, la dirección quedaría de la siguiente manera:

http://192.168.1.127

- La segunda forma de resolver la dirección es mediante una dirección *http* con dominio **.local** basada en el servicio gratuito *Bonjour*, [31] una tecnología de resolución de direcciones basada en DNS sobre IP para descubrir servicios en una red de área local -LAN (*Local Area Network*). Es un servicio desarrollado por *Apple* que, sin embargo, ofrece soporte para otros Sistemas Operativos como *Linux* o *Windows*. De esta manera, para resolver la dirección se deberá introducir en el navegador la siguiente dirección:

http://octopi.local

3.1.4.2 Red WAN (*Wide Area Network*) del sistema de impresión

En este apartado se describe la posibilidad del acceso a la plataforma del servicio de impresión a través de Internet, si bien no es una opción recomendable debido a las vulnerabilidades que esto supondría.

El acceso se realizaría a través de un servicio de DNS Dinámica configurado en el *router* inalámbrico conectado a internet. De esta manera, cualquier ordenador conectado a internet podría acceder a la plataforma web a través de la siguiente dirección:

http://<Dirección IP de la Raspberry Pi>

3.2 Elección del servicio de impresión 3D

Para el presente proyecto se han tenido en cuenta los dos principales servicios de impresión 3D que existen actualmente, Octoprint y Astroprint (Figura 3-8). Realmente este segundo servicio, Astroprint, es un *fork* de Octoprint, es decir, está basado en su código fuente. Para la elección de la plataforma que finalmente se iba a utilizar en el presente proyecto se han realizado sendas pruebas sobre la impresora actual, y si bien ambas resultaron funcionar correctamente, hay ciertos aspectos que han inclinado la balanza hacia el lado de Octoprint. En la siguiente imagen se puede observar una comparativa entre las interfaces de cada uno de los sistemas propuestos.

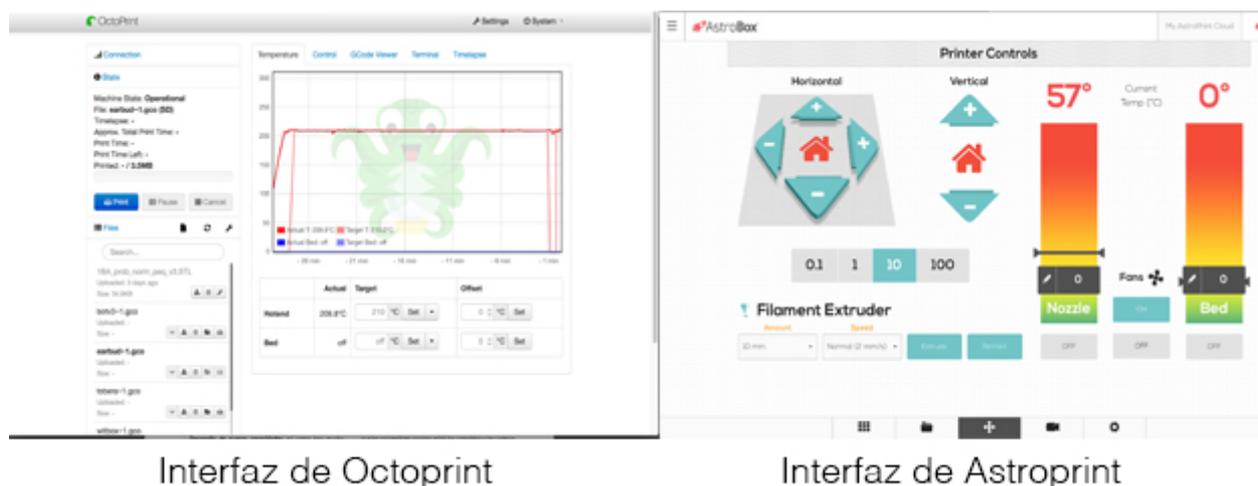


Figura 3-8 Comparativa entre las interfaces de *Octoprint* y *Astroprint*

Basado en el servicio similar que ambas ofrecen, en el apartado visual, ofrece más posibilidades y un entorno más profesional la plataforma Octoprint, lo cual es un factor importante. Sin embargo, es el apartado del desarrollo, la API, la comunidad extendida que apoya al proyecto y por último, la reciente inversión en el sistema por parte de BQ lo que hace que Octoprint sea finalmente la opción elegida en el proyecto.

3.3 Descripción del *software* Octoprint y de su interfaz web

El objetivo del presente apartado es realizar un análisis tanto del software de impresión remota Octoprint, como de la interfaz web que este programa genera.

3.3.1 Descripción del programa Octoprint

El programa Octoprint se ejecuta sobre Raspbian, tal y como se ha descrito anteriormente. Al arrancar la imagen OctoPi sobre la Raspberry Pi, se accede a la consola de comandos y una vez se introduce usuario y contraseña para acceder a la sesión, se puede acceder a la consola de configuración del sistema.

a) En el menú de configuración (Figura 3-9), se nos muestra el siguiente cuadro:

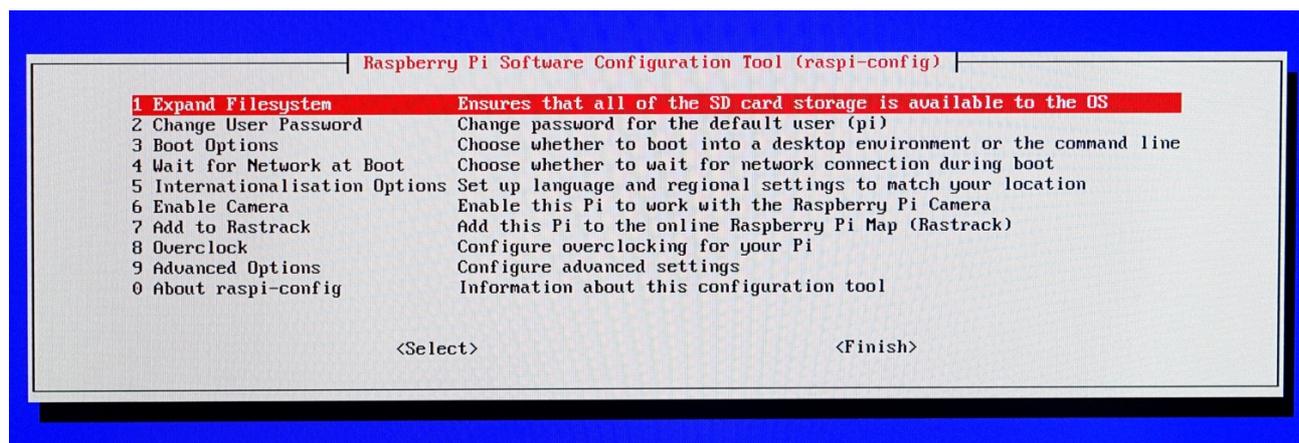


Figura 3-9 Menú de configuración de Octoprint

En la Tabla 2 que se muestra a continuación se describen todas las opciones de configuración mostradas en la Figura 3-9 Menú de configuración de Octoprint

Expand Filesystem	Asigna todo el espacio disponible de la tarjeta SD para almacenamiento del sistema.
Change User Password	Permite cambiar la contraseña del usuario con el que se esté accediendo.
Boot Options	Se permite configurar las opciones de reinicio del sistema.
Wait for Network at Boot	Opción de espera hasta tener conexión a una red durante el inicio del sistema (inicio más lento).
Internationalisation Options	Se permite variar el lenguaje por defecto y la localización.
Enable Camera	Opción para habilitar la cámara nativa de Raspberry.
Add to Rastrack	Opción para añadir la Raspberry al servicio Rastrack. (Actualmente fuera de línea).
Overclock	Opción para forzar la Raspberry Pi para obtener un rendimiento superior.
Advanced Options	Submenú donde se incluyen opciones de configuración avanzadas.
About Raspi-config	Información acerca de la herramienta de configuración.

Tabla 2 Descripción de las opciones de configuración de Octoprint

A través del menú **Advanced Options**, se permite permite acceder a un submenú de opciones de configuración avanzadas como el que se muestra en la Figura 3-10:

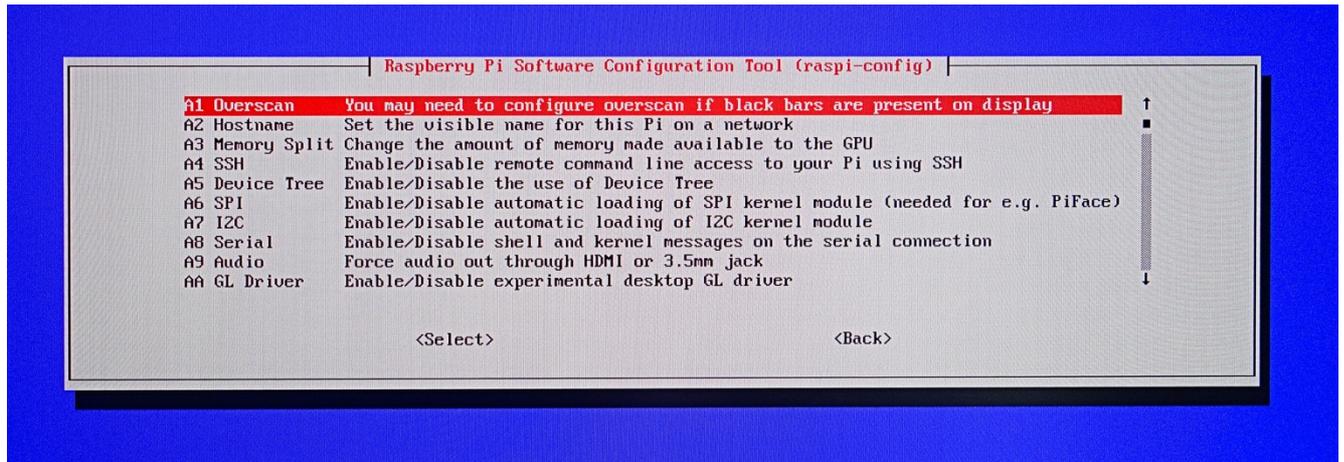


Figura 3-10 Menú del apartado de configuración *Advanced Options*

En la Tabla 3 se describen cada uno de los parámetros de la Figura 3-10.

Overscan	Permite configurar la presentación del sistema y la corrección de errores de visualización
Hostname	Configuración del nombre visible de la Raspberry Pi en una red
Memory Split	Configuración de la cantidad de memoria disponible para la GPU
SSH	Permite abrir/cerrar el puerto 22 para la comunicación vía SSH
Device Tree	Permite activar/desactivar el árbol de dependencias de dispositivos
SPI	Permite activar/desactivar la comunicación por protocolo SPI
I2C	Permite activar/desactivar la comunicación por bus de datos I2C
Serial	Permite activar/desactivar el <i>shell</i> y los mensajes raíz en la conexión serie
Audio	Permite configurar la salida de audio vía jack 3,5mm o HDMI
GL Driver	Permite activar/desactivar un entorno gráfico experimental

Tabla 3 Descripción del sub-menú *Advanced Options*

3.3.2 Descripción de la interfaz web Octoprint

Habiendo seguido los pasos para la instalación del software especificados en el apartado 3.4.2, una vez realizada la conexión entre la Raspberry Pi y el ordenador que se vaya a utilizar para controlar la impresora, mediante los pasos descritos en el apartado 3.4.3, accederemos a la interfaz web de control Octoprint que se muestra en la Figura 3-11.

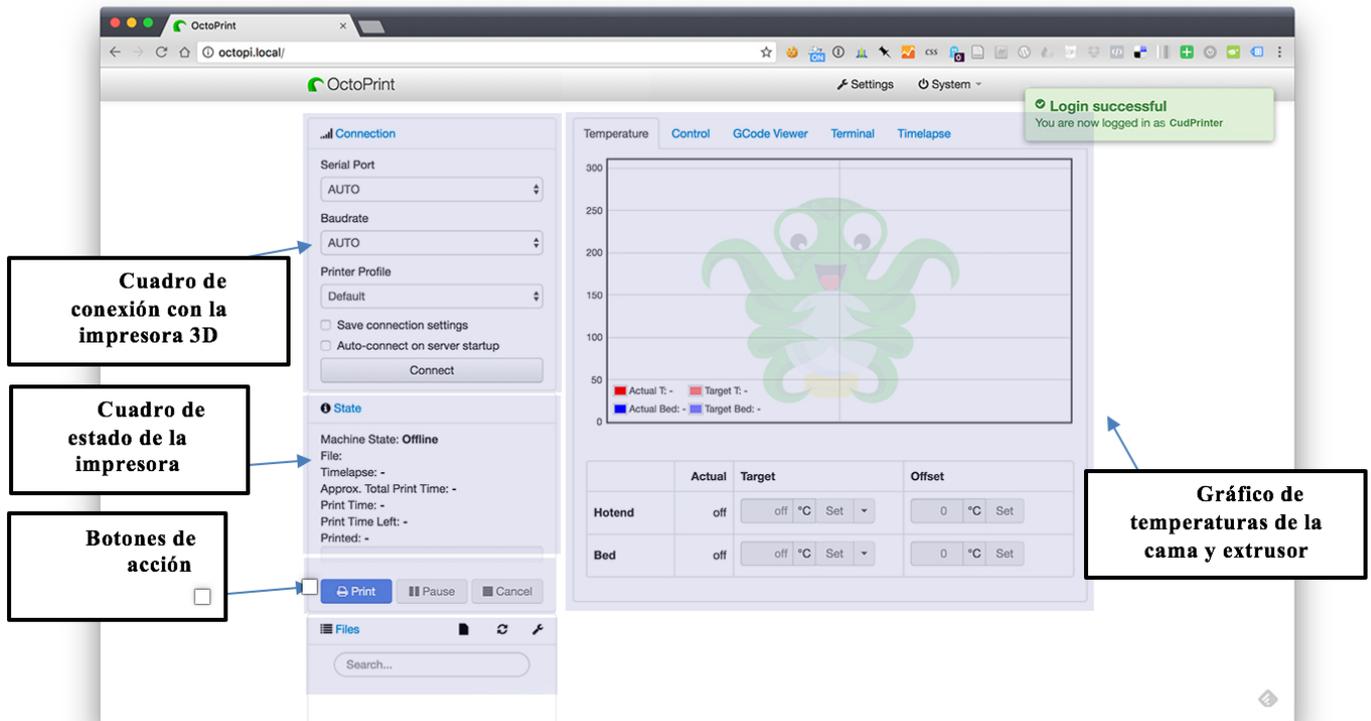


Figura 3-11 Interfaz web de Octoprint

La interfaz incluye una serie de botones que permiten tomar ciertas acciones tales como configurar la conexión con la impresora, acciones sobre la impresora como comenzar, pausar o cancelar la impresión y un panel central donde por defecto encontramos una lectura visual en forma de gráfica de las temperaturas de la cama y del extrusor con respecto al tiempo.

En el apartado **Control**, como se muestra en la Figura 3-12 se permite el control del cabezal de manera manual, actuando directamente sobre los motores así como el control sobre el ventilador, la velocidad de impresión y la velocidad de salida de filamento. En este submenú encontramos la visualización en *streaming* de la cámara y web. Inmediatamente debajo de este cuadro se sitúan distintos botones de acción:

- El eje **X/Y** se encarga de mover el cabezal en sus dos grados de libertad horizontal.
- El eje **Z** se encarga de mover el cabezal en altura, verticalmente.
- El apartado **E** permite extrudir o retraer filamento del extrusor.
- El apartado **General** apaga los motores del cabezal y permite encender o apagar los ventiladores (**Fans on/Fans off**).

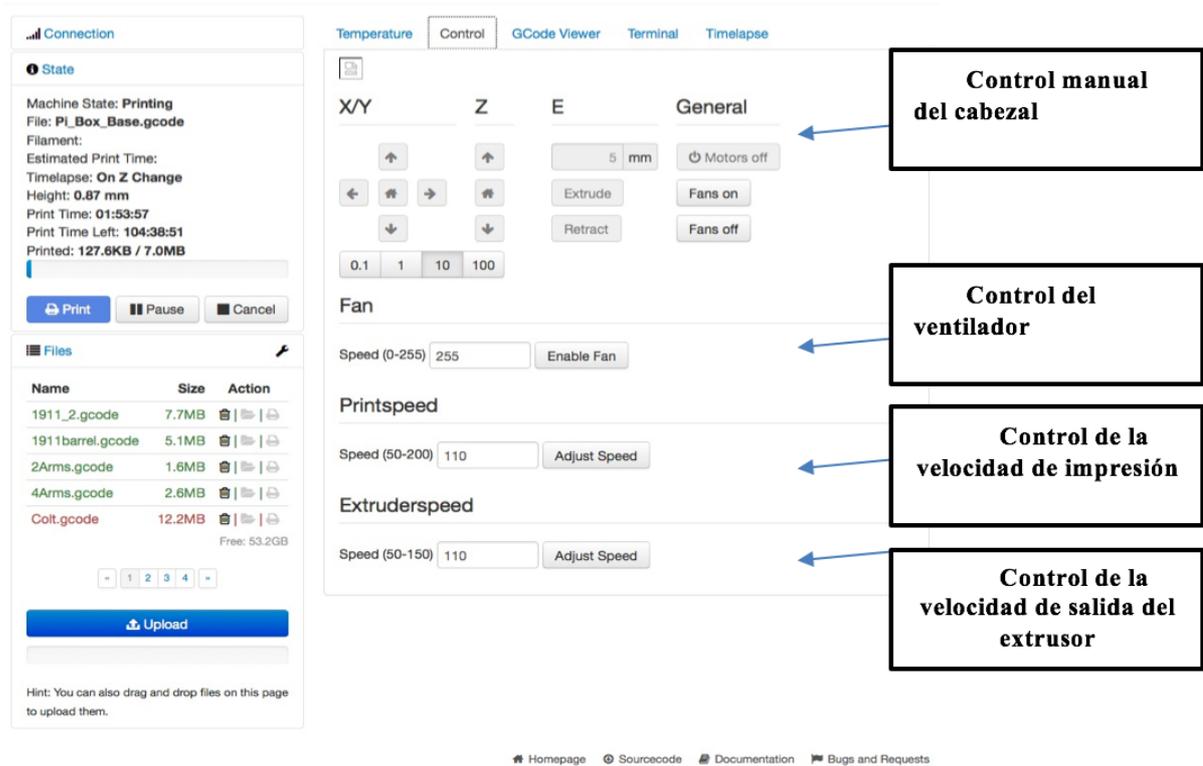


Figura 3-12 Apartado *Control* de Octoprint

En el apartado **GCode Viewer** se accede a un visualizador en tiempo real (Figura 3-13) de las acciones de la impresora cuando imprime piezas en formato *.gco* o *.gcode*. De esta manera se puede analizar el proceso de impresión de la pieza. Sin embargo, este apartado no es funcional cuando se imprimen piezas en formato *.stl*.

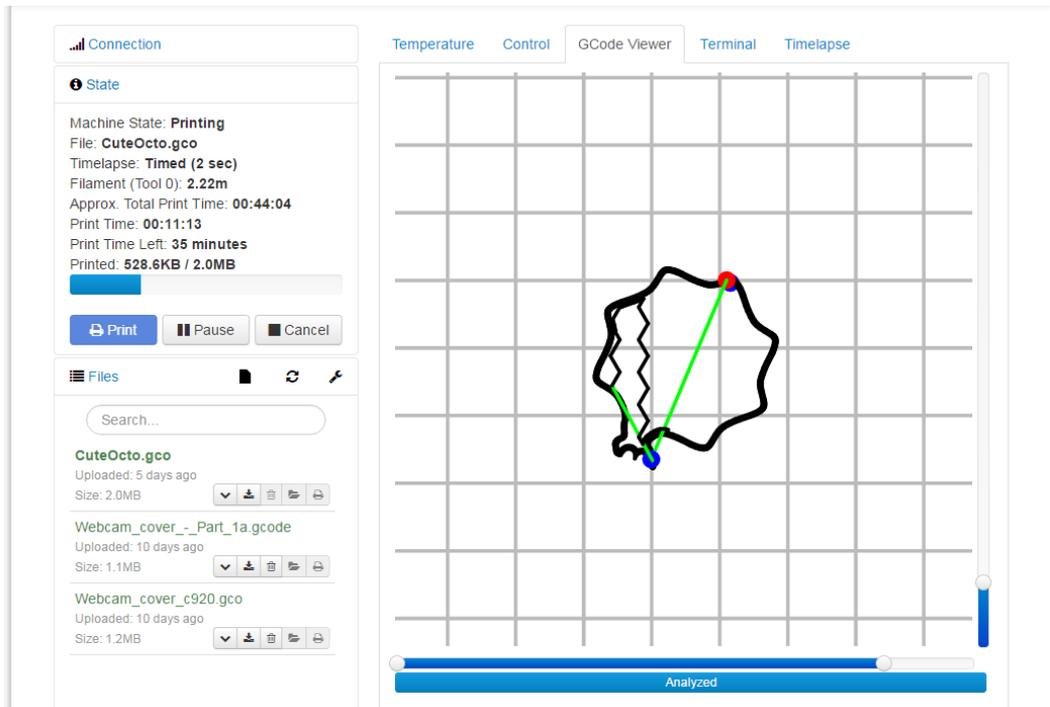


Figura 3-13 Apartado *GCode Viewer* de Octoprint

El siguiente apartado, el botón **Terminal**, permite al usuario leer los comandos que envía la impresora al software y además permite el envío de comandos de acción a la impresora, algo muy útil en apagados de emergencia, por ejemplo. Esto permite una monitorización constante del proceso de impresión, además de mensajes de control para verificar la conexión Octoprint-impresora, se reciben mensajes ACK a cada segundo de impresión. En la Figura 3-14 se muestran los mensajes **Send** que envía Octoprint a la impresora, y los mensajes **Recv**, que en este caso corresponde a la información que proporciona la impresora 3D sobre la cantidad de "información" impresa.

The screenshot shows the Octoprint interface with the **Terminal** tab selected. On the left, the **State** panel indicates the machine is **Printing from SD** with the file **earbud~1.gco (SD)**. The print time is **00:25:26** with **3 hours** left. The **Files** panel shows a list of files including **earbud~1.gco**. The main terminal window displays a log of communication: **Send** commands (e.g., `N1387 M105*20`) and **Recv** responses (e.g., `ok T:208.9 /210.0 B:0.0 /0.0 T0:208.9 /210.0 @:127 B@:0`). Below the terminal, there are checkboxes for **Suppress M105 requests/responses** and **Suppress M27 requests/responses**, and a **Send** button. A hint at the bottom suggests using arrow keys to recall commands.

Figura 3-14 Apartado *Terminal* de Octoprint

El último de los apartados de la interfaz es el botón **Timelapse**. Este apartado, como se muestra en la Figura 3-15, da acceso a la cámara web que está conectada a Raspberry Pi para revisar el proceso de impresión. De esta manera, mediante la configuración de este modo, se puede programar la toma de imágenes de manera secuencial para después, mediante una edición automática, crear un *timelapse*, es decir, un video de un proceso acelerado a partir de esas fotografías.

The screenshot shows the **Timelapse Configuration** panel. Under **Timelapse Mode**, the mode is set to **On Z Change**. There are input fields for **5** Frames/layer and **30** frames/second. A **Save Settings** button is present. Below this, the **Finished Timelapses** section shows a table with the following data:

Name	Size	Action
E3DM_Thimble_L01_s50_20130816073807.mpg	5.1MB	🗑️ 🔗

At the bottom of the table, there are navigation controls showing **1** item.

Figura 3-15 Apartado *Timelapse* de Octoprint

En la configuración de este apartado, el modo **On Z Change** indica que por defecto se tomará una fotografía por cada capa impresa de la pieza, lo cual, dependiendo de la pieza, puede resultar una cifra demasiado baja para crear un efecto de video de calidad. En el apartado de su derecha se pueden configurar las fotografías que se toman por capa impresa y los fotogramas por segundo a los que se reproducirá el video final. De esta manera se crea una estimación de la duración final del video. Para conseguir una fluidez aceptable en el video, por convenio, se recomiendan 24 fps. El video final queda alojado en el servidor web y mediante los botones de acción se puede borrar o descargar a nuestro ordenador.

3.3.3 Descripción de las opciones de configuración de la interfaz web de Octoprint

Desde el apartado **Settings** de Octoprint (Figura 3-16), se puede acceder a la configuración avanzada del sistema, además se pueden crear nuevos perfiles de impresión como se especificará más adelante y por último, incluye un gestor de *plugins*, complementos a la aplicación para aportar al sistema funcionalidades extra.

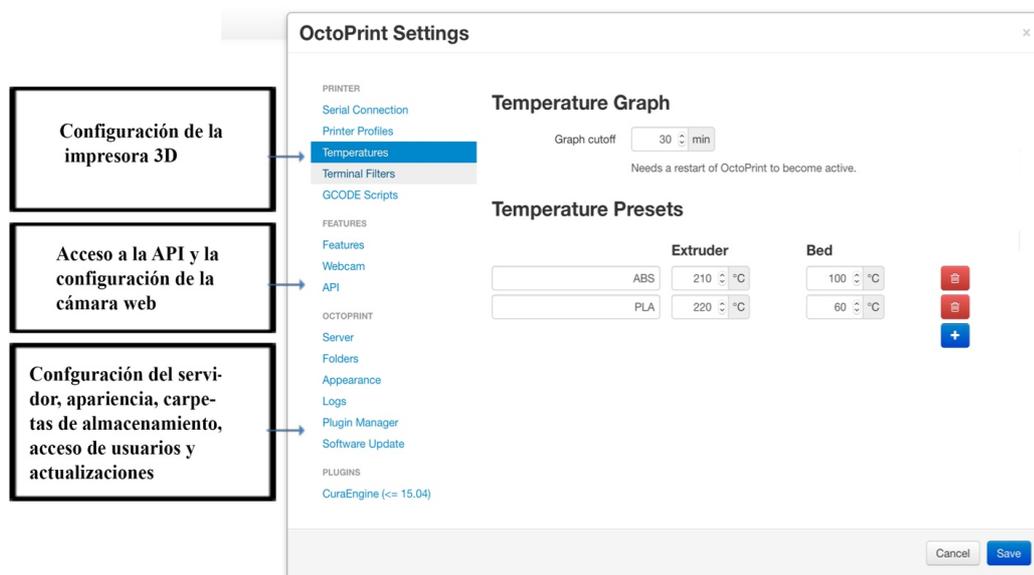


Figura 3-16 Ventana de configuración de Octoprint

El programa posee ciertos automatismos preinstalados; la elección del puerto serie, las temperaturas de la cama y del extrusor y la longitud del eje "tiempo" de la gráfica de temperaturas tienen por defecto valores válidos para operar la impresora, de esta manera se podría empezar a imprimir sin apenas configuración.

El apartado **Printer Profiles** (Figura 3-17) toma gran importancia en el proceso de impresión, nos permite modificar parámetros como el color del filamento o el origen de coordenadas que ha de tomar la impresora. Además permite configurar el número de extrusores así como su diámetro y activar el calentamiento de la cama, algo muy recomendable al imprimir con PLA.

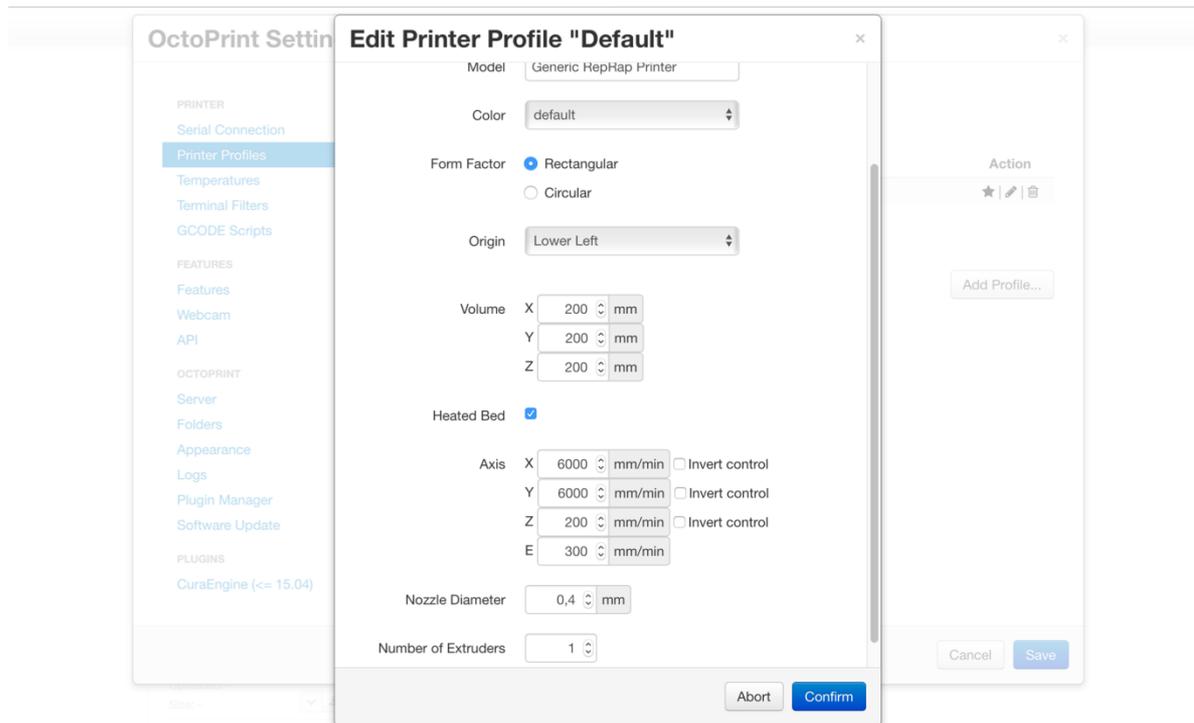


Figura 3-17 Menú desplegable de configuración de los perfiles de impresión

En la pestaña "**Plugin Manager**" se permiten gestionar los *plugins*, en el caso de requerir alguna funcionalidad extra, los *plugins* se pueden buscar en la página oficial de Octoprint [23] y la propia interfaz web incluye un gestor de descargas e instalación.

Para obtener una experiencia más completa en lo relativo a los perfiles de impresión, es conveniente acceder a ellos a través del *plugin CuraEngine 15.04*. Este botón (Figura 3-18), a través del motor de impresión, permite la importación de perfiles de impresión de dicho software.

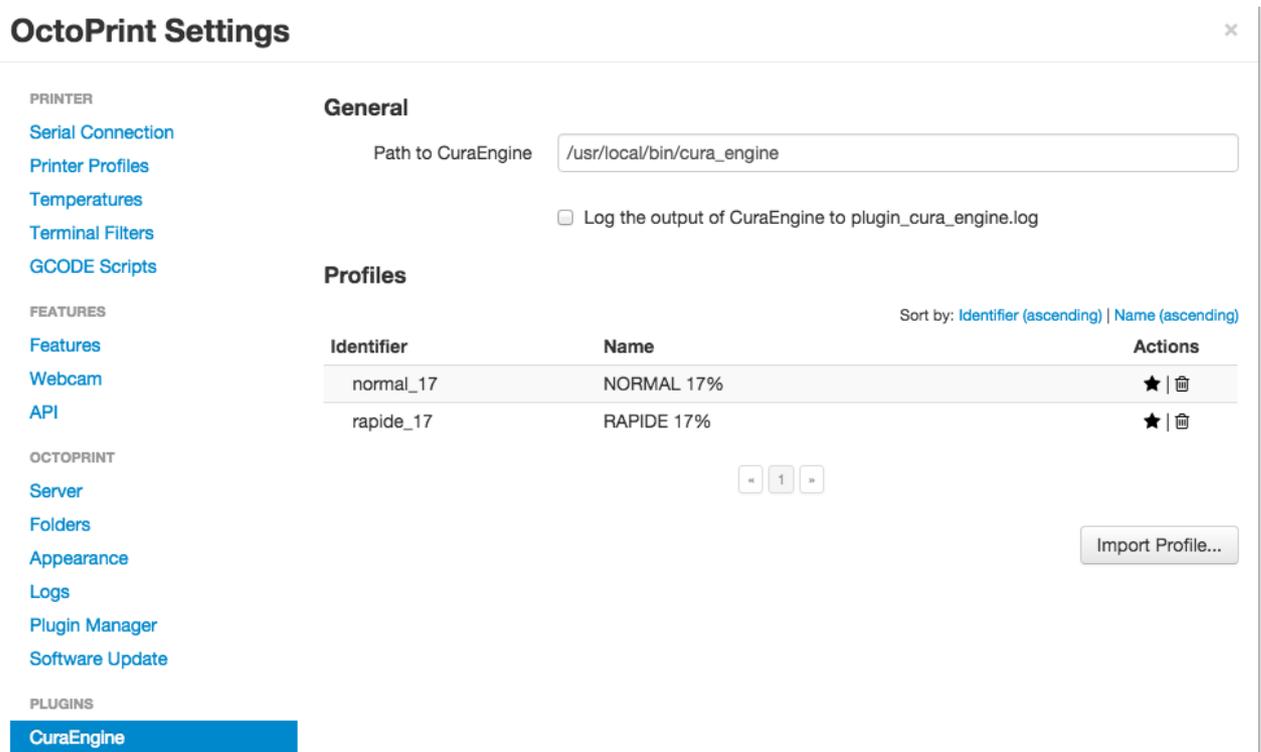


Figura 3-18 Perfiles de impresión de CuraEngine

Los perfiles de impresión de *CuraEngine* permiten modificar los siguientes parámetros a mayores:

- **Altura de capa:** Es lo que comúnmente se conoce como calidad de la pieza, es la medida en altura de cada capa de la pieza, a menor altura de capa, mayor calidad de la pieza como se puede observar en la Figura 3-19.



Figura 3-19 Comparativa de dos piezas con distintas alturas de capa [32]

- **Espesor del diámetro:** Se refiere al espesor de las paredes exteriores, es recomendable dar un valor múltiplo del diámetro del extrusor.
- **Retracción:** Acción de retirar el filamento cuando el extrusor se mueve sin imprimir de una zona a otra. Esta acción evita que hilos de material (Figura 3-20) queden en la pieza final.

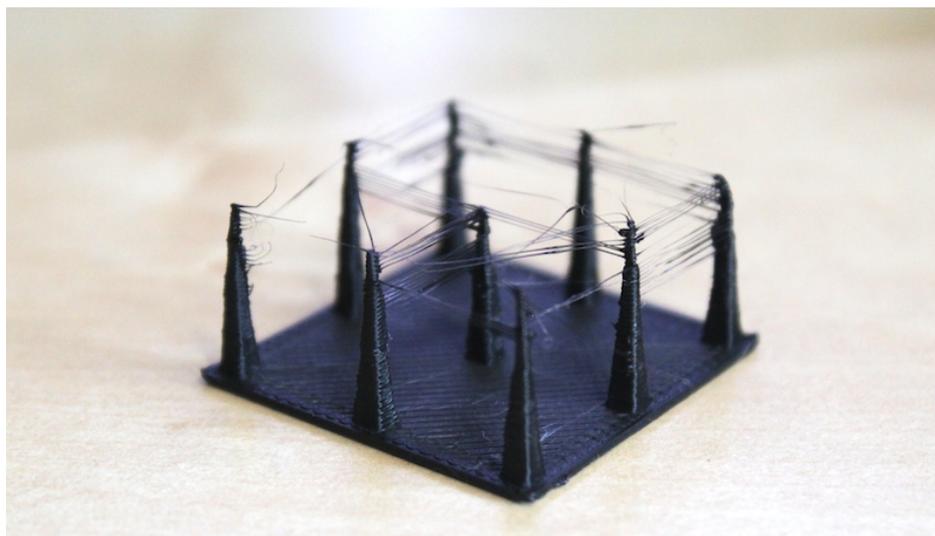


Figura 3-20 Pieza con imperfecciones producto de no utilizar retracción [32]

- **Densidad de Relleno:** Mediante este parámetro se regula la densidad del interior de la pieza, como se puede ver en la Figura 3-21. Un 100% corresponde a una pieza maciza, es recomendable imprimir con valores medios para no disparar los tiempos de fabricación, aunque todo esto depende de las necesidades que se le vayan a requerir a la pieza final.



Figura 3-21 Comparativa de piezas impresas con distinta densidad [32]

En el Título 3.3.2, en el apartado de la configuración de la cámara web, se tienen en cuenta los factores que van a afectar a la toma del video. En cambio, en el apartado **webcam** (Figura 3-22) de la configuración de Octoprint, se permite acceder a otros parámetros que complementan al anterior menú descrito. En la Tabla 4 se especifica el uso de cada apartado.

The screenshot shows the 'OctoPrint Settings' window. On the left, a sidebar lists various settings categories: PRINTER (Serial Connection, Printer Parameters, Temperatures, Terminal filters), FEATURES (Features, Webcam, Cura, Access Control, Api), and OCTOPRINT (Folders, Appearance, Logs). The 'Webcam' option under FEATURES is highlighted in blue. The main content area displays the following settings:

- Stream URL:
- Snapshot URL:
- Path to FFMPEG:
- Timelapse bitrate:
- Enable OctoPrint watermark in timelapse movies
- Flip webcam horizontally
- Flip webcam vertically

At the bottom right, there are 'Cancel' and 'Save' buttons.

Figura 3-22 Apartado webcam de la configuración del sistema

En la Tabla 4, se definen cada uno de los apartado de la Figura 3-22.

Stream URL	Enlace web a la retransmisión de video en directo
Snapshot URL	Enlace web a las capturas de pantalla secuenciales que darán lugar al <i>timelapse</i>
Path to FFMPEG	Ruta donde se alojará el archivo de video generado en streaming en formato MPEG
Timelapse Bitrate	Tasa de bits del <i>timelase</i> , será determinante para la calidad de video.

Tabla 4 Descripción de las opciones de configuración de la cámara web

3.4 Instalación y configuración de Octoprint

La presente guía está basada en la que se ofrece en [23] y en su repositorio de GitHub. Incluye desde la instalación del SO en Raspberry Pi así como las necesidades para conectarse desde un ordenador al servidor web. Además, se incluye una guía de configuración del software para comenzar a imprimir con la impresora en el apartado Configuración de la interfaz web de Octoprint.

3.4.1 Instalación de Octoprint

La instalación del paquete Octopi incluye el SO Raspbian con el *software Octoprint, MPEG streamer* y *CuraEngine*:

1. Descarga la última versión de OctoPi desde la página web de Octoprint desde su página web, [23]. (Figura 3-23)



Getting Started with OctoPi

Figura 3-23 Página web oficial de Octoprint [23]

2. Formatea una tarjeta de memoria MicroSD de al menos 8 GB en formato FAT32. En este caso se ha utilizado en *software* de código libre **SDFormateer** (Figura 3-24), pero existen cantidad de plataformas *open source* con las mismas funcionalidades.

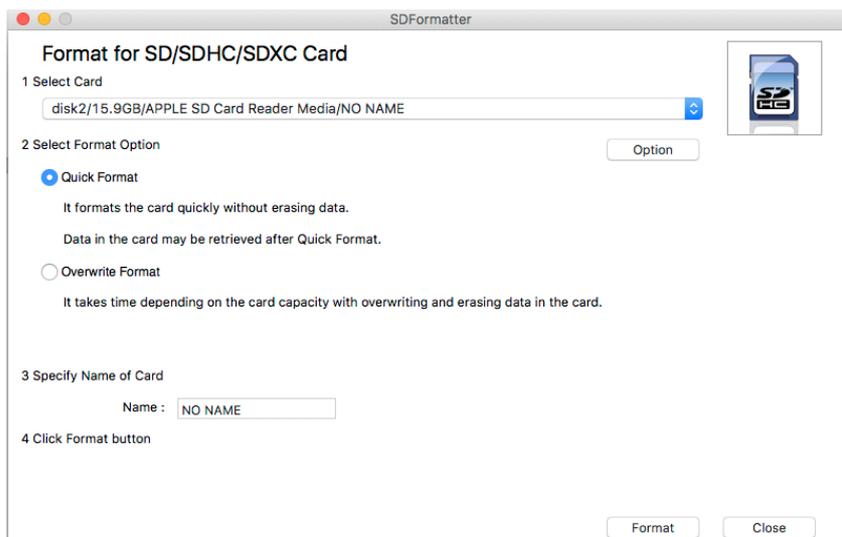


Figura 3-24 Software para formatear dispositivos de almacenamiento (**SDFormateer**)

3. Una vez se haya descargado Octopi, es necesario descomprimir la carpeta y extraer el documento con la extensión *.img*

4. El siguiente paso consiste en instalar el contenedor .img en la tarjeta micro SD. Para instalar una imagen se ha utilizado el software de código libre **ApplePi-Baker** (Figura 3-25), que es válido para ordenador con SO *MacOS*, en el caso de SO *Windows* o *Linux*, es recomendable utilizar el software **WIN32**.

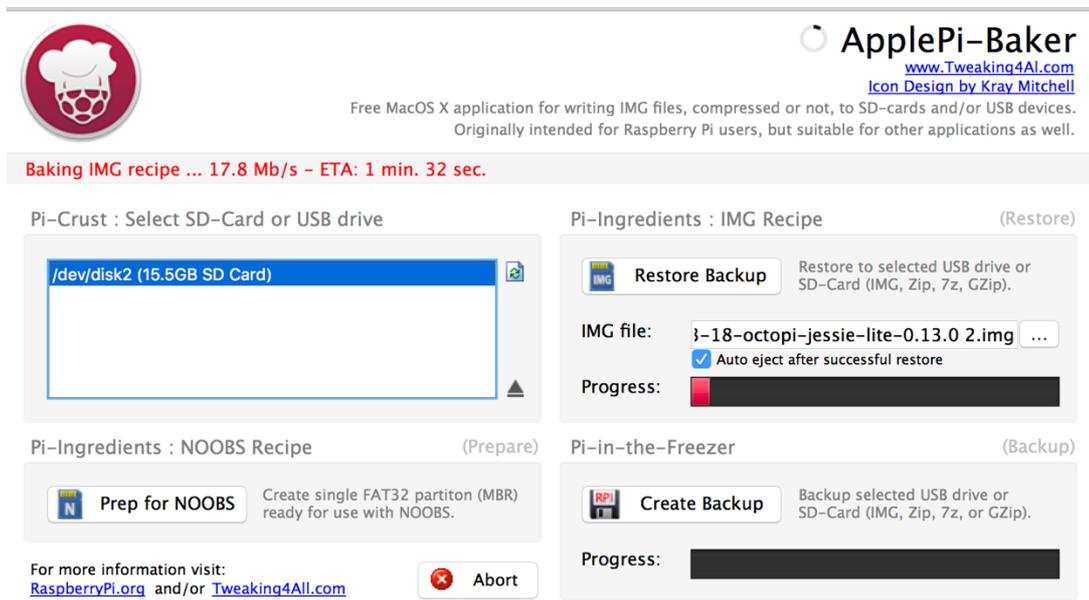


Figura 3-25 Software para grabar imágenes en una tarjeta SD (*ApplePi-Baker*)

5. Una vez instalada la imagen en la tarjeta de memoria, es necesario configurar la red a la que se va a conectar la Raspberry Pi. Esta red debe tener el puerto 22 (SSH) abierto para permitir la conexión entre el ordenador y la Raspberry Pi. La configuración de red se realiza modificando los campos del fichero **octopi-network.txt** dentro de los archivos del sistema. Solamente hay que seguir las instrucciones del texto, completando el nombre de la red y la contraseña, en el caso de que la haya, de la red wifi o por cable a la que nos parámetros mostrados en la figura. Por último hay que retirar las almohadillas de las líneas que vayan a ser código para que el sistema las detecte como tal. (Figura 3-26)

```

octopi-network.txt
# You can use this file to manually set up your network configuration.
#
# This file is included into /etc/network/interfaces, so anything that
# works by editing that file is also possible here.

### WIFI CONFIGURATION #####
# The three segments below should cover you in most cases if you run
# a wifi network that uses either WPA/WPA2 or WEP encryption.
#
# Just uncomment the lines prefixed with a single # of the configuration
# that matches your wifi setup and fill in SSID and passphrase.

## WPA/WPA2 secured
#iface wlan0-octopi inet manual
#   wpa-ssid "wificud"
#   wpa-psk "escuelanaval"

## WEP secured
#iface wlan0-octopi inet manual
#   wireless-essid "put SSID here"
#   wireless-key "put password here"

## Open/unsecured
#iface wlan0-octopi inet manual
#   wireless-essid "put SSID here"
#   wireless-mode managed

### WIRED CONFIGURATION #####
# The following segment allows you to configure your wired connection
# with a static IP.

```

Figura 3-26 Documento de texto donde se almacenan los parámetros de red de Octoprint

6. Hecho esto, se procederá a insertar la tarjeta microSD a la Raspberry Pi, posteriormente se conectará la placa a la corriente y se arrancará Raspberry Pi. Al visualizar en pantalla el arranque de la placa Raspberry Pi, se mostrará una interfaz similar a la de la Figura 3-27



```

[ OK ] Reached target Basic System.
Starting LSB: Change pi's password via /boot/octopi-password.txt...
Starting LSB: Change pi's hostname via /boot/octopi-hostname.txt...
Starting Regular background program processing daemon...
[ OK ] Started Regular background program processing daemon.
Starting dhcpcd on all interfaces...
Starting Login Service...
Starting LSB: webcam daemon...
Starting LSB: OctoPrint daemon...
Starting LSB: Autogenerate and use a swap file...
Starting LSB: triggerhappy hotkey daemon...
Starting oshbi mSATA/SD Stack...
Starting D-Bus System Message Bus...
[ OK ] Started D-Bus System Message Bus.
[ OK ] Started Oshbi mSATA/SD Stack.
Starting System Logging Service...
Starting Permit User Sessions...
[ OK ] Started Restore Sound Card State.
[ OK ] Started LSB: Change pi's password via /boot/octopi-password.txt.
[ OK ] Started LSB: Change pi's hostname via /boot/octopi-hostname.txt.
[ OK ] Started LSB: webcam daemon.
[ OK ] Started LSB: OctoPrint daemon.
[ OK ] Started LSB: triggerhappy hotkey daemon.
[ OK ] Started Permit User Sessions.
[ OK ] Started LSB: Autogenerate and use a swap file.
[ OK ] Started Login Service.
[ OK ] Started dhcpcd on all interfaces.
[ OK ] Reached target Network.
Starting OpenSSH Secure Shell server...
[ OK ] Started OpenSSH Secure Shell server.
Starting HAProxy Load Balancer...
Starting /etc/rc.local Compatibility...
[ OK ] Reached target Network is Online.
Starting LSB: Start NTP daemon...
My IP address is 192.168.1.100
[ OK ] Started /etc/rc.local Compatibility.
Starting Wait for Plymouth Boot Screen to Quit...
Starting Terminate Plymouth Boot Screen...
[ OK ] Started HAProxy Load Balancer.
[ OK ] Started Wait for Plymouth Boot Screen to Quit.
[ OK ] Started LSB: Start NTP daemon.
[ OK ] Started Terminate Plymouth Boot Screen.
Starting Getty on tty1...
[ OK ] Started Getty on tty1.
Starting Serial Getty on ttyAMA0...
[ OK ] Started Serial Getty on ttyAMA0.
[ OK ] Reached target Login Prompts.
[ OK ] Reached target Multi-User System.
[ OK ] Reached target Graphical Interface.
Starting Update UTMP about System Runlevel Changes...
[ OK ] Started Update UTMP about System Runlevel Changes.

Raspbian GNU/Linux 8 octopi tty1
octopi login: _

```

Figura 3-27 Interfaz de usuario de Raspbian

Es conveniente remarcar el hecho de que Octoprint no tiene entorno gráfico de escritorio ya que no está pensado para ser operado desde la Raspberry Pi. El acceso al menú de configuración se realizará a través de la consola de comandos.

7. El primer paso una vez dentro de Raspberry Pi es introducir el nombre de usuario y contraseña. En Raspbian por defecto estos son **"pi"** y **"raspberrypi"** respectivamente. Una vez validados, estaremos dentro de nuestra sesión de Raspbian con Octoprint instalado.

3.4.2 Configuración de Octoprint

En este apartado se abordará tanto la configuración de Octoprint desde Raspbian como la configuración de la interfaz web para imprimir con la impresora BQ Witbox descrita en el título 3.1.3.

1. En primer lugar, para acceder al menú de configuración se ha de introducir en el *shell* el siguiente comando: `sudo raspi-config`. Una vez dentro de este menú, se nos mostrará la siguiente pantalla: (Figura 3-28)

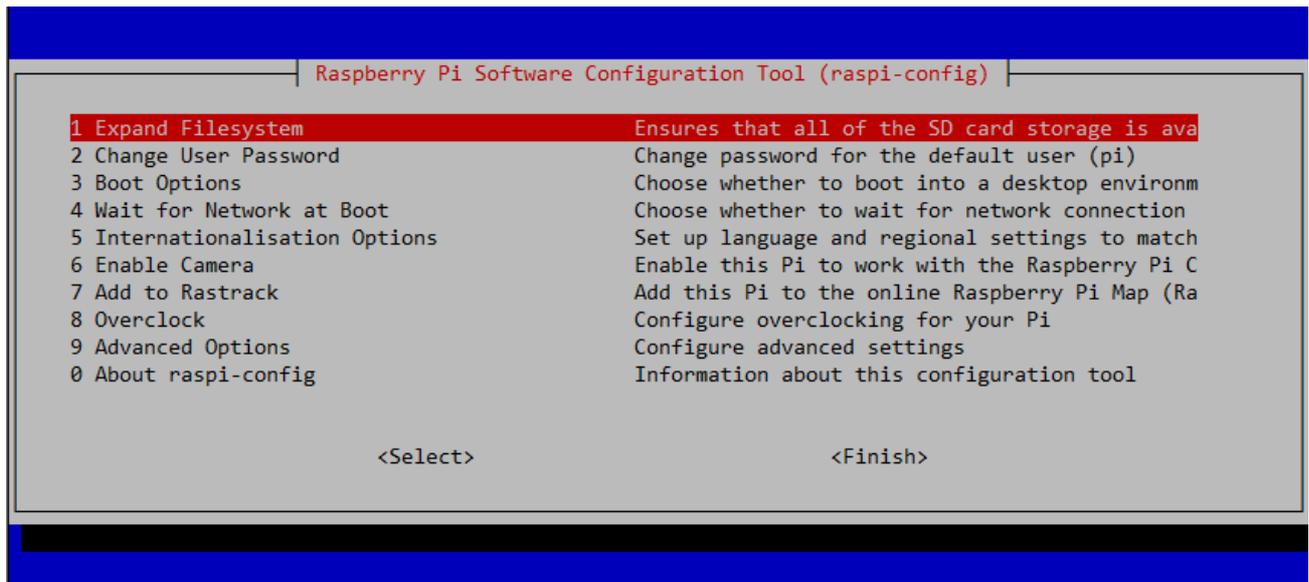


Figura 3-28 Menú de configuración de Octoprint

2. Una vez dentro, el primer paso que hay que realizar es expandir el Sistema de Ficheros mediante el primer botón del menú. Esto destinará la memoria de la tarjeta microSD a *Raspbian*.
3. Como medida de seguridad activa, para tener acceso restringido a este servidor, es recomendable cambiar la contraseña. Esto se realiza mediante el segundo botón del menú "**Change User Password**".
4. Para que el teclado sea reconocido de manera correcta por el sistema, es necesario configurar el desplegable número 5 "**Internationalisation Options**". Una vez dentro, deberemos seleccionar el tipo de teclado que tengamos. (Figura 3-29)

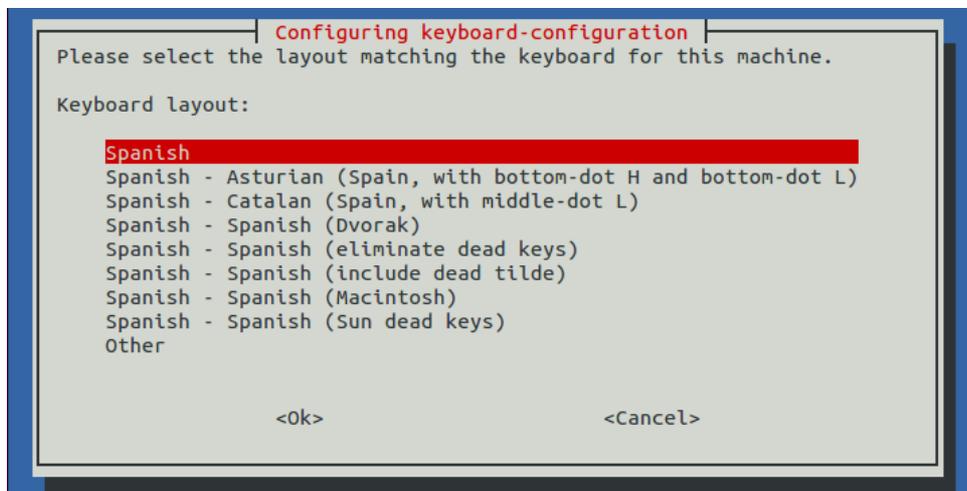


Figura 3-29 Panel de configuración del teclado en Octoprint

5. Una vez realizado este paso, accederemos al menú de configuración avanzada mediante el botón número 9, "**Advanced Options**" (Figura 3-28) Simplemente se deberán habilitar los apartados A4 "**SSH**", que permite la comunicación SSH a través de la red entre varios dispositivos conectados a esa red, y el A8, "**Serial**", que permitirá la comunicación con la impresora a través de la conexión serial por bus de datos. (Figura 3-30).

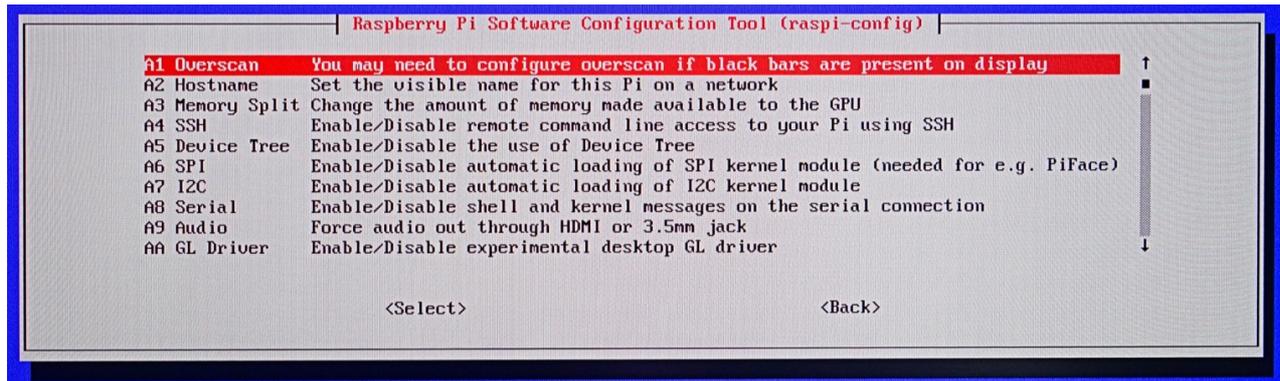


Figura 3-30 Submenú "Configuración avanzada"

6. Realizados estos pasos, el proceso de configuración de Octoprint habrá concluido y el sistema estará listo para imprimir.

3.4.3 Configuración de la interfaz web de Octoprint

El primer paso es acceder al servidor web que ha creado Octoprint, esto se podrá realizar desde cualquier ordenador conectado a esa red mediante la introducción en un navegador web la URL <http://octopi.local> o bien <http://<dirección IP asignada a Raspberry Pi>>. Una vez dentro, se nos mostrará el siguiente entorno: Figura 3-31

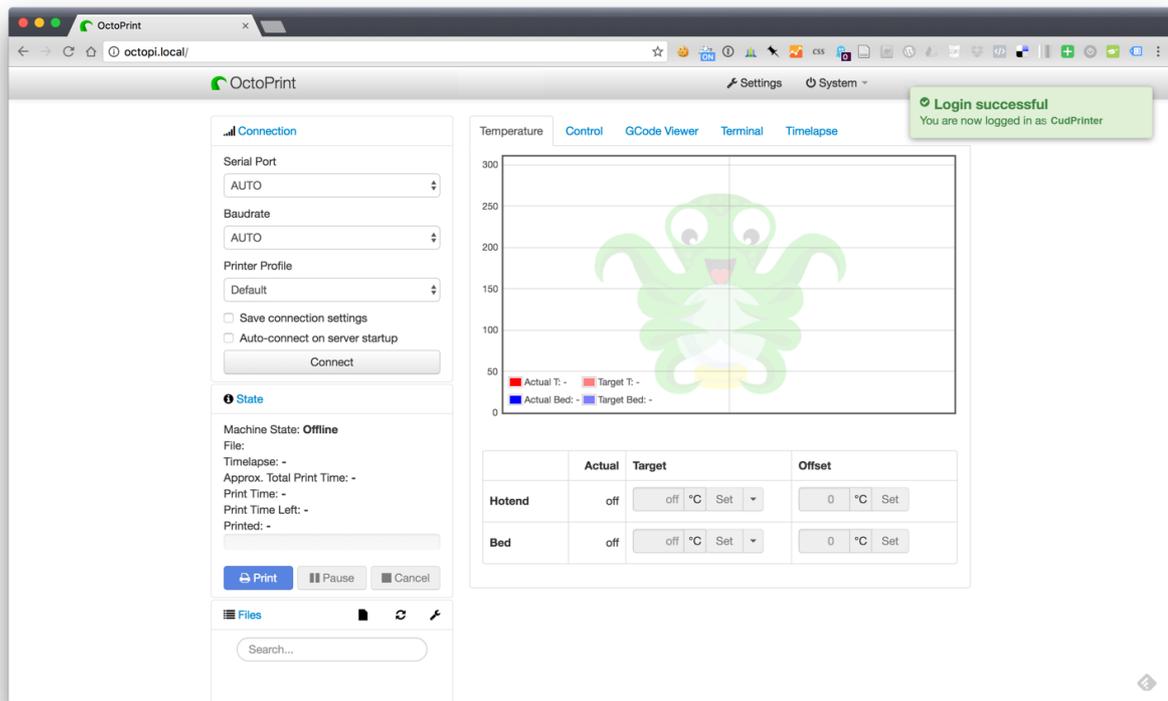


Figura 3-31 Entorno gráfico de Octoprint

El siguiente paso es conectar la impresora, para esto, en el cuadro "**Connection**" seleccionaremos un "**Serial Port**" AUTO y un "**Baudrate**" de 11500, siendo este un valor conservador de la tasa de actualización de datos en la comunicación interfaz web-impresora 3D. En el selector del perfil de impresora, se mantendrá en "default" para la asignación automática del perfil. Cuando se realice la conexión se observará el siguiente menú de conexión: (Figura 3-32)

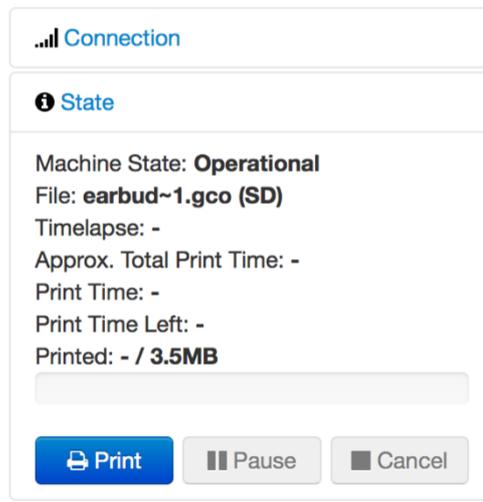


Figura 3-32 Cuadro de conexión con la impresora 3D en Octoprint

A partir de este momento, ya existe conexión con la impresora, es hora de configurar el apartado de "Control" donde se tendrá acceso a la cámara web y el control manual del cabezal.

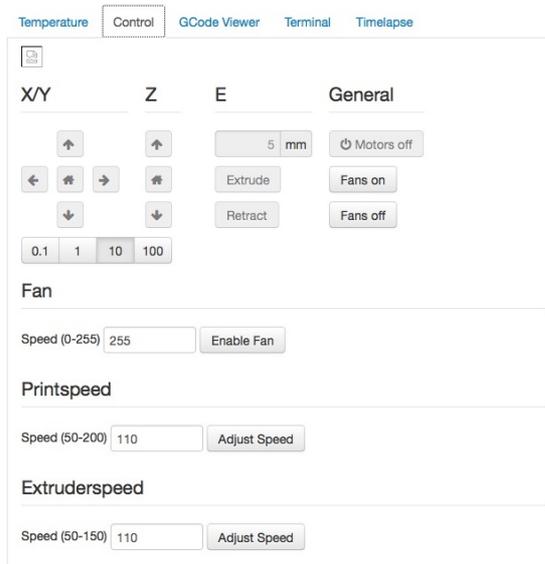


Figura 3-33 Apartado Control de la interfaz web

En esta ventana la única configuración necesaria consiste en hacer *click* sobre el botón "home" (Figura 3-34) para sincronizar los movimientos y poder establecer el origen de coordenadas de la impresora.



Figura 3-34 Botón home

Como paso de control, se abrirá el menú "Terminal" para comprobar que existe una conexión correcta con la impresora 3D, tal y como se expone en la sección 3.3.2. Una vez comprobada la conexión, es necesario abrir la pestaña "settings" de Octoprint (Figura 3-35).

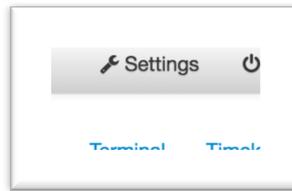


Figura 3-35 Pestaña *Settings* de Octoprint

En este apartado se comenzará configurando las temperaturas de impresión, es decir, las temperaturas que alcanzará el extrusor dependiendo del material de impresión en cada caso (PLA o ABS) . Para esta impresora, como se puede ver en la Figura 3-36, las temperaturas serán 210°C para PLA y 220°C para ABS. [30]

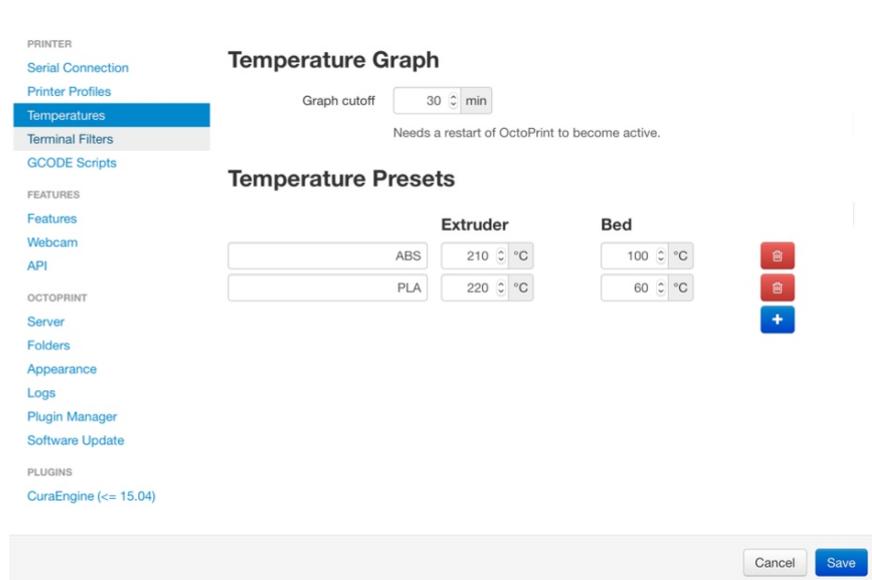


Figura 3-36 Apartado de configuración de temperaturas de impresión en Octoprint

Más adelante, en la pestaña del menú **plugins**(Figura 3-36) se debe hacer *click* sobre **CuraEngine 15.04**, desde el menú desplegable que aparece se puede realizar una gestión de los perfiles de impresión de *CuraEngine*, de esta manera se imprimirá con la calidad deseada para cada pieza. Por último se guardarán los cambios realizados usando el botón **Save** de la esquina inferior derecha y la impresora estará lista para comenzar a imprimir (Figura 3-36).

Si se desea ampliar las funcionalidades de Octoprint, esta plataforma pone a disposición de sus usuarios un repositorio de *plugins*, herramientas adicionales desarrolladas sin ánimo de lucro. En el apartado **Plugin Manager** (Figura 3-37), se permite gestionar los diferentes *plugins* mediante un buscador e instalador. Además, la página web de Octoprint, [23] ofrece un repositorio de *plugins* donde se pueden consultar las funcionalidades de cada uno. Desde la fuente oficial se enlaza a la web Github (www.github.com) donde se incluye el código de cada *plugin* así como una guía de configuración.

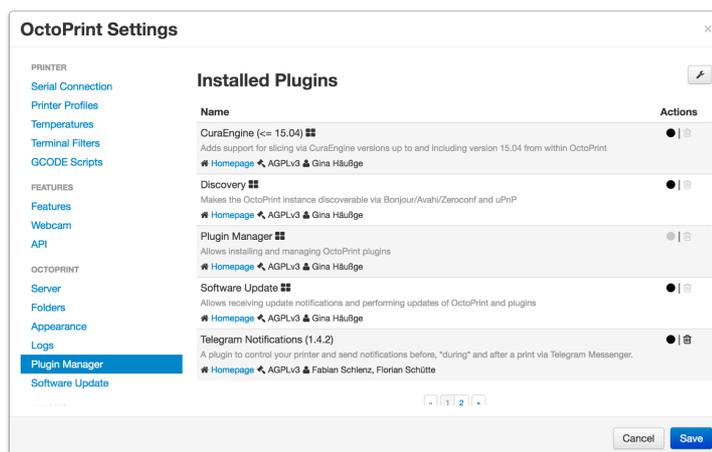


Figura 3-37 Ventana del Plugin Manager de Octoprint

3.5 Creación de un bot de Telegram

Telegram Messenger (Figura 3-38) es una aplicación de mensajería instantánea vía Internet, desarrollado en el año 2013 por Nikolai y Pavel Durov. La compañía, actualmente, es gestionada por una Fundación del mismo nombre que opera en Berlín. Las aplicaciones están disponibles en 10 sistemas operativos, tanto para ordenador como para dispositivos móviles. [33]



Figura 3-38 Logo de Telegram Messenger [33]

En los siguientes apartados se van a tratar las funcionalidades que aporta al sistema la creación de un bot sobre esta plataforma. Además se realiza una descripción del proceso de instalación y configuración del mismo.

3.5.1 Descripción del bot de Telegram creado.

Existe en el repositorio de *plugins* de Octoprint un *plugin* para integrar el software Octoprint con la aplicación de mensajería Telegram. Esto es posible gracias a los *bots*, un *software* que "imita" el comportamiento humano en el sentido de que envía respuestas vía chat en función de los comandos que se le envíen.

Para el caso del *bot* creado para el control y supervisión de la impresora, este permite que a través de comandos se puedan realizar acciones; las acciones implementadas inicialmente se describen en la Tabla 5.

Comando	Función
abort	Aborta la impresión en curso
shutup	Desactiva las notificaciones automáticas
dontshutup	Reactiva las notificaciones automáticas
status	Muestra el estado así como una fotografía
settings	Configuración actual de notificaciones y permite modificarla
files	Muestra la lista de las piezas disponibles precargadas en la SD
print	Permite comenzar una impresión. Se requiere confirmación
togglepause	Pausar/reaundar la impresión en curso
con	Conectar/Desconectar impresora.
upload	Permite cargar una pieza en gcode para su impresión
sys	Muestra los comandos del sistema Octoprint
ctrl	Usa controles definidos de Octoprint
tune	Configuración de la tasa de flujo de filamento y temperatura
user	Proporciona información de usuario
help	Se proporciona la lista de comandos disponibles

Tabla 5 Comandos del bot de Telegram

Dichas acciones están contenidas en un código en lenguaje Python que se nutre de las librerías y API que Telegram pone a la disposición de desarrolladores. Una vez en la aplicación de Telegram, todos estos comandos se encuentran precargados y acompañados de su descripción.

El bot que se ha desarrollado puede encontrarse en el buscador de Telegram Messenger como **CudPrinter** (Figura 3-39) y se le puede mencionar como `@cudPrinter_bot`. Es un *bot* público que, sin embargo, requiere de permisos que gestiona el administrador del *bot* para permitir a un usuario realizar acciones sobre la impresora.



Figura 3-39 Búsqueda del bot "CudPrinter" en Telegram

3.5.2 Creación del bot de Telegram

El proceso de creación de un *bot* de Telegram se realiza en la propia aplicación, en primer lugar se ha de buscar al *bot* "Botfather", el mecanismo que Telegram utiliza para los *bots* que se utilizan en su plataforma. Una vez encontrado, se recibe el siguiente mensaje: (Figura 3-40).

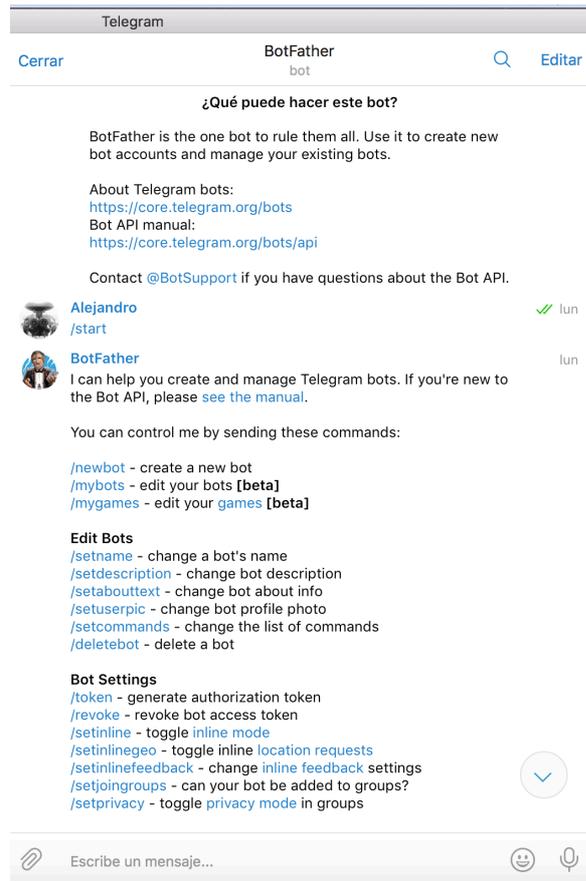


Figura 3-40 Presentación inicial de "botfather" en Telegram

Al enviar el comando `/newbot`, se crea un nuevo *bot* y mediante los comandos `/setname` y `/setuserpic`, se puede elegir el nombre y la imagen de perfil del nuevo *bot*. Una vez se ha creado el bot, el servidor devuelve un *token* de acceso para enlazar el *bot* con el servicio sobre HTTP. Este token es un código alfanumérico que posteriormente servirá para modificar el bot en la API de Telegram. Por motivos de seguridad, se ha omitido en la Figura 3-41

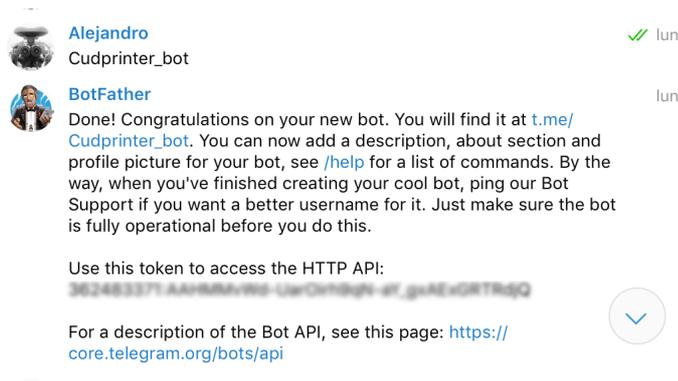


Figura 3-41 Devolución del *token* de acceso del *bot*

La siguiente tarea será definir la lista de comandos para los cuales, el *bot* devuelve una respuesta. Para esta operación se teclea el comando `/setcommands` (Figura 3-42). En el caso del bot creado, la lista de comandos es la de la Tabla 5.

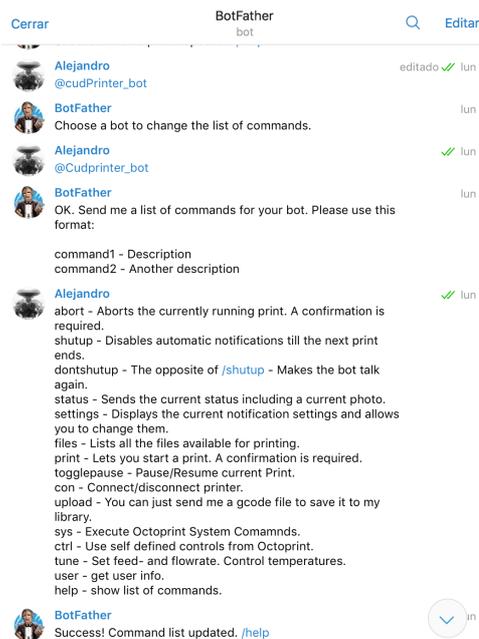


Figura 3-42 Envío de comandos al bot de Telegram

Una vez realizados estos pasos, el bot está creado y es hora de instalar el *plugin* sobre el *software* Octoprint.

3.5.3 Instalación y configuración del plugin de Telegram en Octoprint

La instalación del *plugin* (Figura 3-43) se realiza mediante el procedimiento descrito en la sección 3.4.3 y una vez instalado se procede a la configuración donde se enlazará a Octoprint con Telegram y se gestionarán los usuarios con acceso al *bot*. En el siguiente submenú de la configuración se introduce el *token* que se especifica en el apartado 3.5.2.

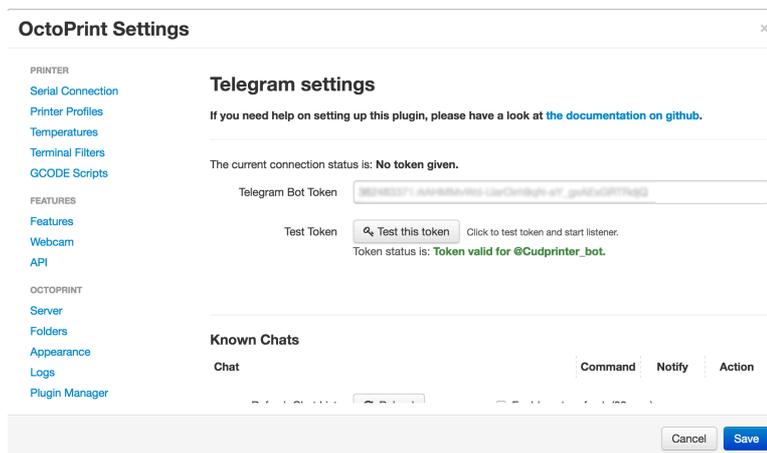


Figura 3-43 Configuración del plugin de Telegram para Octoprint

El último paso merece una constante revisión, ya que se trata de la autorización a los usuarios que acceden al bot para interactuar con la impresora. La visualización de dicho usuarios de Telegram se realiza desde este mismo submenú y las ventanas que gestionan los permisos son las mostradas en la Figura 3-46 y la Figura 3-47. Además, desde la aplicación de Telegram, cada usuario puede consultar estos datos introduciendo el comando `/user`. (Figura 3-44)

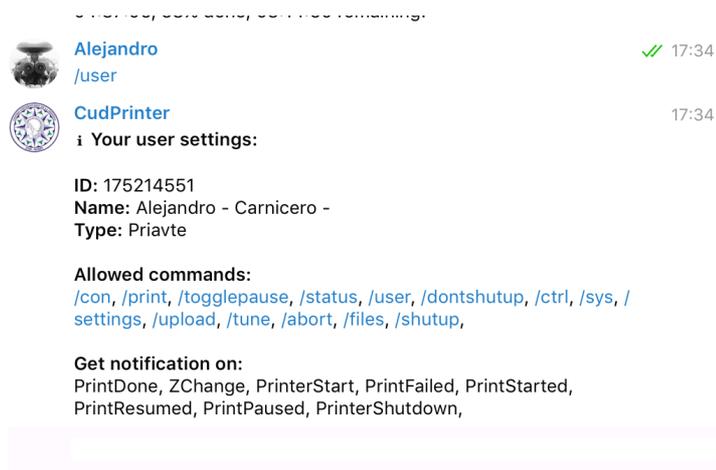


Figura 3-44 Permisos de usuario sobre el *bot* desde la aplicación Telegram

3.5.4 Valor añadido y conclusión

Teniendo en cuenta las funcionalidades extra que proporciona este *plugin*, especialmente en el aspecto de la monitorización y la conectividad, demuestra ser una herramienta que aporta valor añadido, mostrando una buena integración con el sistema y una alta estabilidad. Proporciona principalmente, una capacidad de monitorización pasiva a tiempo completo, lo cual es importante a la hora de conocer de antemano cual es el producto final que se va a obtener, permitiendo cancelar o modificar la impresión si se advierte algún tipo de modificación de los factores. Además, es importante remarcar el hecho de que a través de este *plugin* proporciona acceso vía Internet mediante cualquier dispositivos móvil *iOS* o *Android*.

3.6 Seguridad de la plataforma remota de impresión 3D

Aunque la Raspberry Pi esté conectada al ordenador en el que se está llevando la impresión en Red Local, Octoprint permite su operación a través de Internet mediante esta aplicación. Esto, en términos de seguridad es muy importante; de esta manera se concluyen dos vías de entrada para obtener accesos sobre la impresora:

- A través de la Red Local generada por el router con clave WPA2 como se especifica en la Figura 3-6. Así, solo los usuarios con acceso a esta clave tendrán acceso a la plataforma. Además, sobre el *software* Octoprint, se permite el control de acceso por medio de usuario y contraseña. Lo cual es una medida de seguridad a mayores. (Figura 3-45)

⚠ Configure Access Control
 Please read the following, it is very important for your printer's health!
 OctoPrint by default now ships with Access Control enabled, meaning you won't be able to do anything with the printer unless you login first as a configured user. This is to **prevent strangers - possibly with malicious intent - to gain access to your printer** via the internet or another untrustworthy network and using it in such a way that it is damaged or worse (i.e. causes a fire).
 It looks like you haven't configured access control yet. Please **set up an username and password** for the initial administrator account who will have full access to both the printer and OctoPrint's settings, then click on "Keep Access Control Enabled":
 Username
 Password
 Confirm Password
Note: In case that your OctoPrint installation is only accessible from within a trustworthy network and you don't need Access Control for other reasons, you may alternatively disable Access Control. You should only do this if you are absolutely certain that only people you know and trust will be able to connect to it.
Do NOT underestimate the risk of an unsecured access from the internet to your printer!
 Disable Access Control Keep Access Control Enabled

Figura 3-45 Control de acceso del software Octoprint

- A través de Internet mediante el *bot* de Telegram si se obtienen los permisos de visualización y escritura sobre dicho *bot*. Actualmente y para las pruebas se ha dado pleno acceso a dos usuarios, sin embargo, el sistema soporta tantos como se requiera. Además, se dispone de un menú para dar acceso personalizado a cada usuario. Consecuentemente, se pueden definir dos tipos de permisos: de acción y de visualización. (Figura 3-46, Figura 3-47)

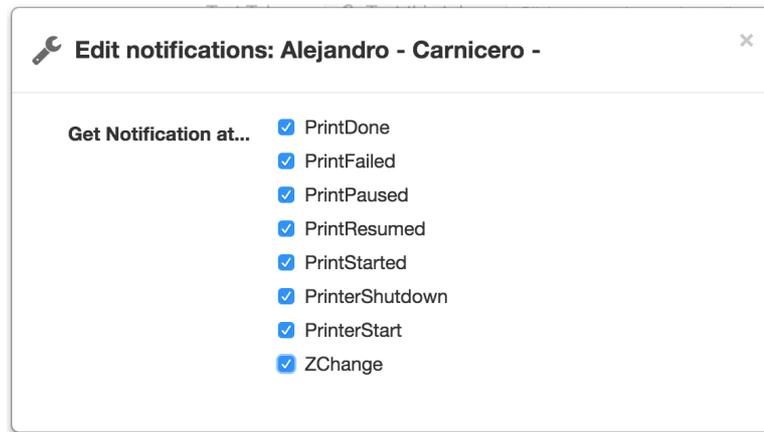


Figura 3-46 Permisos de visualización del *bot*

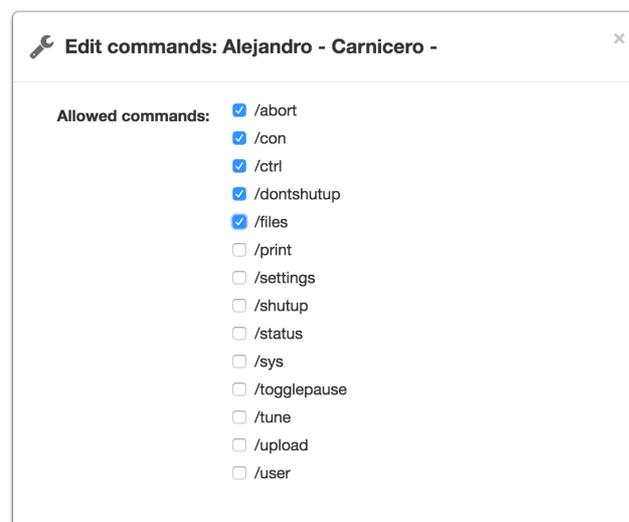


Figura 3-47 Permisos de acción del *bot*

Por medio de estas dos vías de acceso, además de la forma obvia, de manera manual a través de la impresora se restringe la operación de la impresora únicamente a los usuarios que el administrador autorice a su uso. Además, Telegram es famoso por su sistema de cifrado en tiempo real, lo cual convierte a la plataforma en un sistema todavía más robusto en términos de seguridad.

3.7 Desarrollo de un sistema auxiliar para la plataforma de impresión 3D

3.7.1 Introducción y características de la plataforma de sistemas auxiliares

Debido a la posibilidad de que la impresora 3D opere en una sala aislada sin supervisión humana se ha creído necesaria la implementación de ciertos sistemas que aporten cierta autonomía y seguridad al sistema de impresión. De esta manera, la independencia a la hora de operar la impresora 3D a distancia se hace todavía más real creando una experiencia de usuario más completa.

Basado en los recursos disponibles y en las necesidades que han surgido, se ha creído conveniente desarrollar un sistema con los siguientes requisitos:

- Capacidad de operar de forma autónoma
- Estabilidad del sistema durante largos periodos de uso.
- Facilidad de operación y alta fiabilidad de los subsistemas para evitar fallos en la impresión derivados de errores ocurridos durante la impresión
- Capacidad de operar de manera independiente de la impresora.
- Procesamiento de datos procedentes de lecturas en pines analógicos y digitales.
- Sencillez en la configuración y en la instalación

En cuanto a las capacidades con las que se ha creído conveniente dotar al sistema, se incluyen las siguientes:

- Sistema de detección de falta de filamento para evitar que la impresora opere sin material.
- Sistema de iluminación automático en función de la luz ambiente
- Sistema de acondicionamiento de la caja de impresión para que la impresora opere en valores de temperatura óptimos.

Debido a los requisitos y capacidades arriba expuestos, convierte a la placa Arduino en una plataforma idónea para este proyecto, si bien existen otras placas con mayor capacidad de procesamiento y por un precio similar, como puede ser Raspberry Pi, la plataforma Arduino cuenta con mayor compatibilidad con periféricos y el entorno de programación es más sencillo; además la capacidad de procesamiento de esta placa resulta más que suficiente para el alcance del presente proyecto.

3.7.2 Requisitos de la placa controladora del sistema auxiliar

Para la elección de la placa más adecuada para el proyecto se han establecido los siguientes requisitos:

- Al menos 4 pines digitales.
- Al menos 2 pines analógicos.
- Conexión puerto serie con un ordenador.
- Disponer de pines extra para futuras mejoras.
- Disponer de memoria suficiente para almacenar el programa.
- Alta estabilidad en largos periodos de uso continuado.

3.7.3 Diseño del sistema auxiliar de la plataforma de impresión 3D

Como controlador del sistema auxiliar se ha optado por la placa Arduino Uno, ya que para las dimensiones de este proyecto no era necesario utilizar sus homólogas superiores como Arduino Mega ya que esta cumple con solvencia los requisitos impuestos en la sección 3.7.2.

En cuanto a los periféricos, tanto de entrada como de salida, se han elegido componentes sencillos, al alcance de cualquiera pero que sin embargo, cuentan con una resolución (en el caso de los sensores) y una fiabilidad que cumple con los mínimos impuestos para el desarrollo del sistema. La arquitectura del proyecto sigue una lógica sencilla de entrada-salida, es decir, cada una de las entradas del sistema, genera una acción en las salidas si se cumplen las condiciones impuestas. El esquema básico del sistema se muestra en la Figura 3-48.

Las entradas del sistema se componen de un una resistencia variable con la luz o LDR (*Light Dependant Resistance*) para medir la luminosidad del entorno, un sensor de temperatura para medir la temperatura del interior de la impresora y por último, un prototipo de detector de falta de filamento que se ha desarrollado desde cero.

En lo relativo a las salidas, se ha instalado una antorcha LED para iluminar el entorno de impresión, un ventilador que hace las veces de acondicionador del entorno de impresión y por último,

se actuará mediante la placa Arduino en la alimentación de la impresora en el caso de tener una detección del falta de filamento.

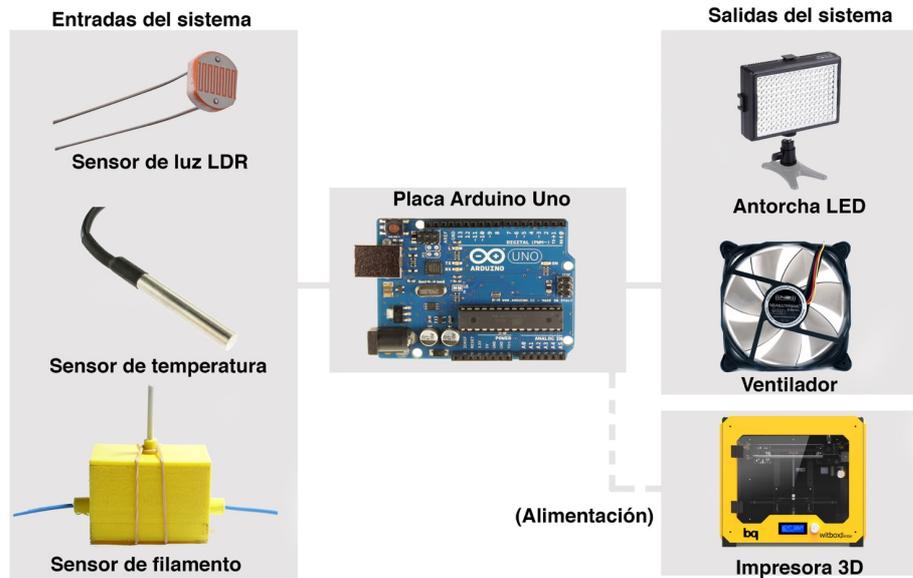


Figura 3-48 Descripción del *hardware* utilizado en el sistema auxiliar

3.7.4 Descripción de los componentes del sistema auxiliar de impresión 3D

En cuanto a los elementos de hardware empleados en el sistema se hacen constar los siguientes:

- **Arduino Uno:** Es una placa con diferentes opciones de conectividad así como diversos pines digitales y analógicos basada en el microcontrolador ATmega328P como "cerebro" del sistema. Este elemento sirve de gestor de todos los sistemas de seguridad y acondicionamiento.

En la Figura 3-49 se describen las diferentes conexiones que ofrece la placa Arduino Uno:

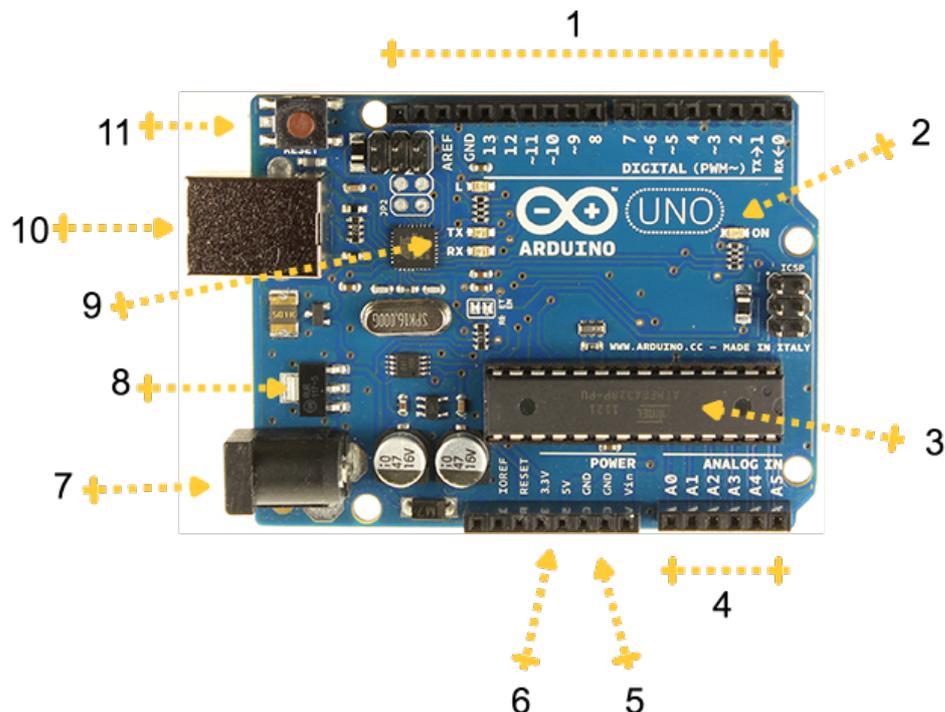


Figura 3-49 Partes de la placa *Arduino Uno* [29]

1. **Pines digitales:** Pueden actuar como entradas o salidas devolviendo los valores booleanos 0 o 1. Estos valores corresponden a una tensión de 0V o 5V. Los pines acompañados de un \sim son pines PWM, es decir, pueden controlar el voltaje de salida mediante pulsos.
2. **LED de encendido:** Se trata de un LED verde que indica que la placa está encendida.
3. **Microcontrolador:** Es un microcontrolador ATmega328P, actúa como controlador de toda la placa.
4. **Pines analógicos:** son capaces de reconocer diferentes voltajes y devolver un valor digital gracias al convertidor analógico-digital de la placa. Pueden ser utilizados como entradas o salidas.
5. **Tomas de tierra de la placa:** se reconocen mediante la inscripción **GND**.
6. **Salidas de tensión de la placa:** Estas proporcionan una caída de tensión de 5 o 3,3 V.
7. **Toma de alimentación:** mediante esta entrada se alimenta a la placa con una tensión que puede variar entre 6 o 12V.
8. **Regulador de tensión:** este componente regula la tensión de entrada a 5V. Tiene una limitación a 20 V pero es aconsejable no exceder 12V.
9. **LED's de TX/RX:** estos LEDs se encienden cuando la placa recibe o transmite datos a través de la comunicación serie.
10. **Entrada USB:** esta entrada permite la carga de programas en la placa conectada a un ordenador. Además proporciona la tensión necesaria para el funcionamiento de la placa.
11. **Botón RESET:** al pulsar este botón, el programa cargado en la placa se reinicia.

- **Sensor digital de temperatura:**

Un **sensor digital de temperatura DS18B20** (Figura 3-50) para obtener las lecturas del interior de la impresora. Sus características se definen en el Anexo I. La recepción de datos se realiza mediante el protocolo **1-wire** con Arduino.



Figura 3-50 Sensor de temperatura digital

- **Antorcha LED**

La antorcha LED que se ha decidido utilizar está dividida en módulos que se pueden conectar en serie, los cuales contienen 5 LEDs de alta intensidad (Figura 3-51). Sus características se pueden encontrar en el Anexo I



Figura 3-51 Tira de LEDs utilizada en el sistema

- **Relés tipo "keyes" Songle 5V DC**

Con el objeto de controlar mediante señales de 5V circuitos de más voltaje, se han utilizados dichos relés cuyas características se recogen en el Anexo I. (Figura 3-52)



Figura 3-52 Relé tipo "keyes"

- **Ventilador DC**

A modo de climatizador, se ha utilizado un ventilador *Artic-Cooling* ARCTIC F9 cuyas características se encuentran en el anexo I. (Figura 3-53)



Figura 3-53 Ventilador de hasta 12V

- **Resistencias LDR**

Con el objeto de medir la luz ambiente del local donde se encuentre la impresora 3D, se ha utilizado un foto-resistor cuyas características se recogen en el Anexo I. (Figura 3-54)

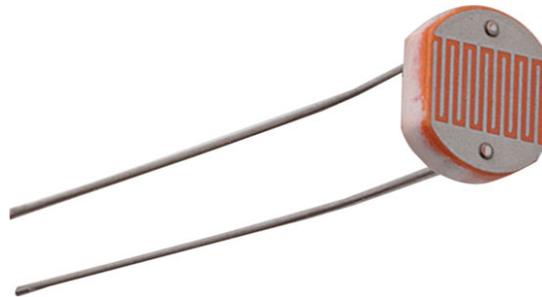


Figura 3-54 Foto-resistor

3.7.5 Descripción del montaje del sistema auxiliar de la plataforma de impresión 3D

En el prototipo que se ha realizado del sistema auxiliar, como se puede observar en la Figura 3-55 Topología de las conexiones del sistema auxiliar de la plataforma de impresión 3D, todos los periféricos se han instalado sobre la misma placa Arduino, de manera que cada uno de los programas se ha almacenado en una función y se ejecutarán en paralelo.

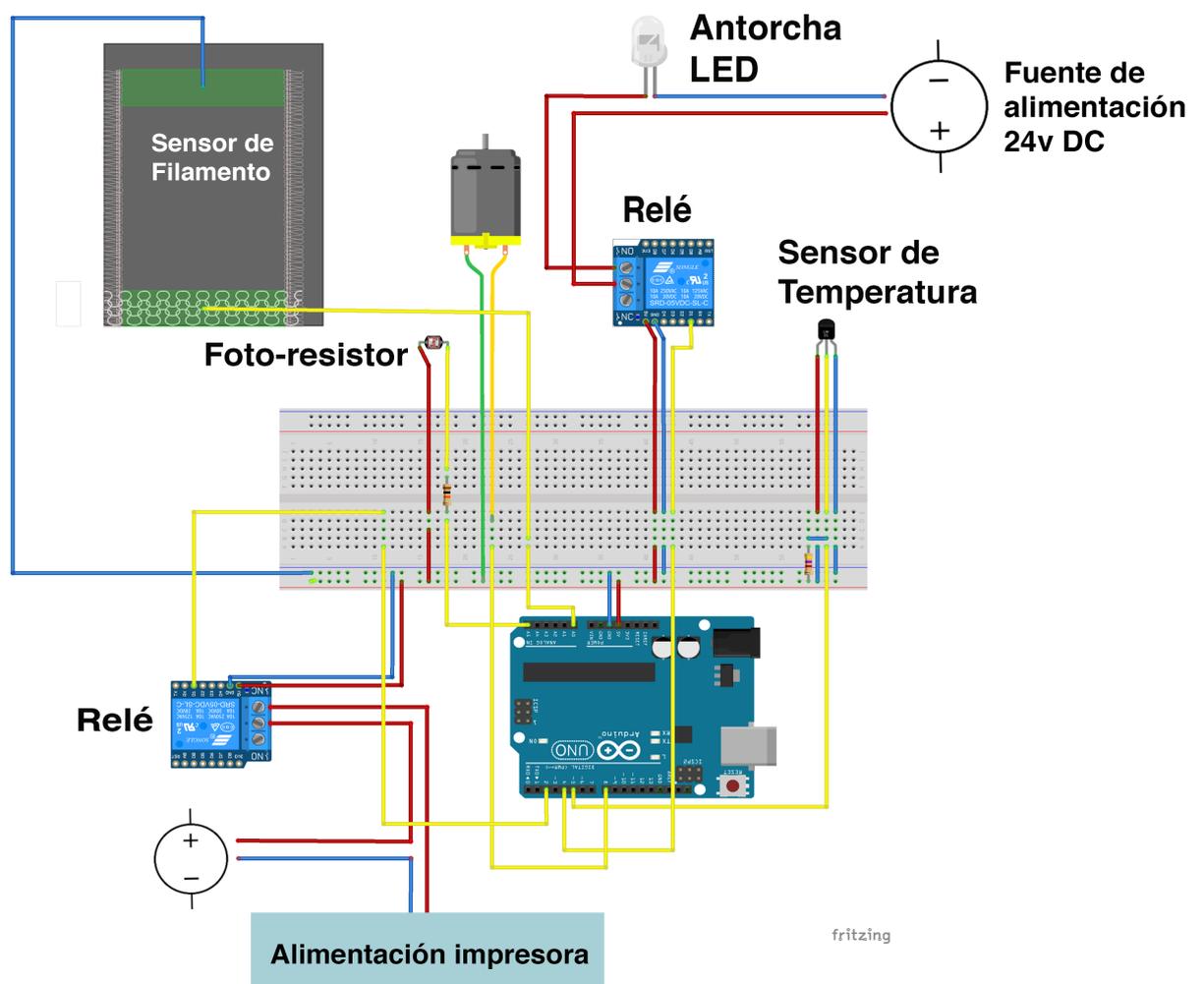


Figura 3-55 Topología de las conexiones del sistema auxiliar de la plataforma de impresión 3D

Como se puede observar en la Figura 3-55, se han alimentado las líneas de la placa de prototipado rápido con las líneas de tierra (azul) y 5 Voltios (rojo). De esta manera se alimenta al motor de DC, los relés, el foto-resistor y el sensor de temperatura. La alimentación de la antorcha LED se realiza a través de una toma de 24 V DC que se conmuta mediante uno de los relés y la alimentación de la impresora es dirigida a través de un relé (parte inferior izquierda de la Figura 3-55 Topología de las conexiones del sistema auxiliar de la plataforma de impresión 3D) para poder controlarla a través de la placa Arduino.

- **Entradas del sistema:**

- Para recibir la lectura del sensor de temperatura se ha utilizado el pin digital 5 como **input**, de esta manera se leen los valores de temperatura en grados Centígrados.

- Para conmutar el relé que controla la alimentación de la antorcha LED, se ha utilizado el pin digital 4 como **output**, de esta manera se puede controlar la posición NO-NC del relé.

- En el caso de la lectura del valor de entrada del foto-resistor, se ha utilizado el pin analógico 5 como **input**. Mediante la variación en el voltaje de entrada se obtiene el valor de luminosidad.

- En la recepción de la lectura del contacto del sensor de filamento, se ha utilizado el pin analógico 5 como **input**. Este pin recibe la lectura del contacto "fijo" del detector.

- **Salidas del sistema:**

- Para conmutar el relé que controla la alimentación de la antorcha LED, se ha utilizado el pin digital 4 como **output**, de esta manera se puede controlar la posición "NO-NC" del relé.

- En el caso de la alimentación del motor del ventilador, se ha decidido hacerlo a través de un pin digital como **output**, el número 8. De esta manera se alimenta con 5V al motor cuando se requiere.

- El relé utilizado para la conmutación de la alimentación de la impresora se conmuta mediante el pin digital número 3 como **output**. Cuando se recibe un "1" en el relé, el relé conmuta al circuito "NO".

3.7.6 Sistema de acondicionamiento de temperatura del espacio de impresión.

3.7.6.1 Motivación del sistema

En el manual de usuario de la impresora 3D [30], se establece que los valores óptimos para imprimir con PLA han de mantenerse en el rango de 15°C y 25°C. En la práctica se ha comprobado que la temperatura es notablemente superior, de esta manera pueden proliferar ciertos problemas como el llamado *warping*, que está relacionado con la falta de adherencia de las primeras capas a la cama, esto produce superficies curvas que deberían ser planas. (Figura 3-56)



Figura 3-56 Pieza con un fallo de "warping" [32]

Otro problema derivado de las altas temperaturas y un gradiente positivo de temperatura acusado, es el derrumbamiento de las superficies en voladizo, la superficie tarda más de lo debido en solidificarse y se producen deformaciones en la pieza, como se puede observar en la Figura 3-57. [34]



Figura 3-57 Piezas en voladizo impresas sin refrigeración [32]

3.7.6.2 Diseño del sistema

Motivado por estos hechos, se ha diseñado un sistema de acondicionamiento para mantener unos valores de temperatura óptimos en el que se sigue una estructura de bucle cerrado, es decir, manteniendo una comprobación constante de los valores de temperatura en la caja de impresión y ejerciendo un control positivo sobre la misma mediante el uso de un ventilador que desaloja calor de la caja. El montaje del sistema donde se integran el ventilador y el sensor de temperatura, se muestra en la Figura 3-58.

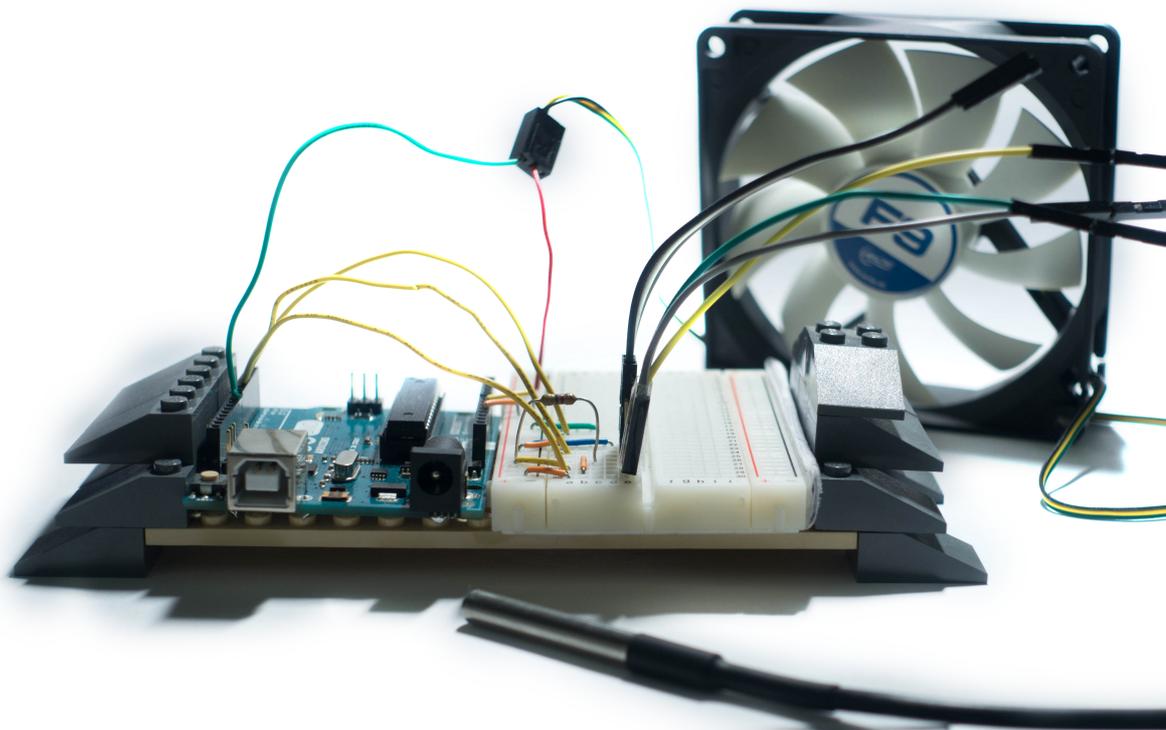


Figura 3-58 Componentes del sistema de acondicionamiento del sistema de impresión

El sistema no necesita alimentación externa a mayores de la propia alimentación de la placa Arduino dado que el resto de componentes se alimentan con los 5V que proporciona la placa.

El sensor de temperatura, conectado a una entrada analógica, recoge a intervalos de 1 segundo la temperatura captada, y es en función de esta, cuando el ventilador, representado en la imagen como un motor de DC, se alimenta o se mantiene en *stand-by* por la salida digital con 5V. En el caso de conectar la placa a un ordenador, existe una comunicación abierta con el puerto serie que muestra la temperatura que recoge el sensor en grados °C para tener una comprobación de que el sensor funciona correctamente. A continuación, en la Figura 3-59, se muestra un esquema aislado del montaje de sistema de acondicionamiento de temperatura:

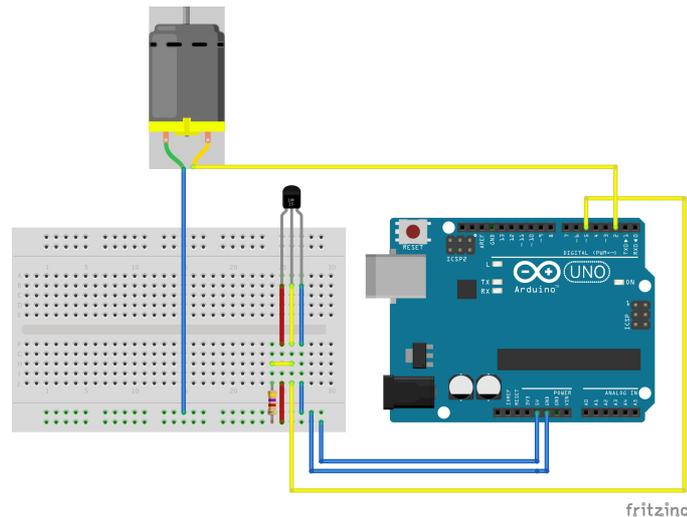


Figura 3-59 Esquema del montaje del sistema de acondicionamiento de temperatura

3.7.6.3 Software del sistema de acondicionamiento de temperatura

La programación de la placa se ha realizado sobre la aplicación nativa de Arduino, un software que además de facilitar el proceso de programación en C++, facilita la gestión de placas, puertos y librerías. Además, existe la ventana *serial* (en la esquina superior derecha de la Figura 3-60), donde se permite la comunicación bilateral con la placa. A través de esta aplicación se carga el programa a la placa para su uso. Si bien este código se ha desarrollado como un sistema aislado del resto, para su presentación final, el programa se ha incluido como función en el programa genera, el cual cargará finalmente en la placa, de esta manera se facilita la lectura del código y se mantiene un orden.

```
#include <OneWire.h> //Se importan las librerías
#include <DallasTemperature.h>

#define Pin 5 //Se declara el pin donde se conectará la DATA

OneWire ourWire(Pin); //Se establece el pino de conexión

DallasTemperature sensors(&ourWire);

int fan=2;
int estado = LOW;
float temp = 0;

void setup() {
  delay(1000);
  Serial.begin(9600);
  sensors.begin(); //Se inician los sensores
  pinMode(fan, OUTPUT);
}

void loop() {
  sensors.requestTemperatures(); //Se solicita la temperatura
  temp = sensors.getTempCByIndex(0);
  Serial.print(temp); //Se lee e imprime la temperatura en grados Celsius
  Serial.println(" grados Centigrados");

  delay(1000); //Se provoca un lapso de 1 segundo antes de la próxima lectura
  if (temp > 24){
    estado = HIGH;
  } else if (temp < 17){
    estado = LOW;
  }
}

Guardado.
El Sketch usa 5802 bytes (17%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.
Las variables Globales usan 263 bytes (12%) de la memoria dinámica, dejando 1785 bytes para las variables locales. El máximo es 2048 bytes.
```

Figura 3-60 IDE de Arduino durante la programación del sistema de acondicionamiento.

En lo relativo a las librerías utilizadas, se han añadido las librerías *DallasTemperature* y *1-Wire*, [29] ambas para facilitar el proceso de lectura de la temperatura mediante el sensor descrito. Así el valor medido se imprime directamente como una temperatura en grados Celsius sin necesidad de realizar cálculos aritméticos. El código comentado se adjunta en el Anexo II.

En cuanto al programa, a continuación se muestra el diagrama de flujo que muestra el algoritmo básico (Figura 3-61) que sigue el sistema de acondicionamiento de temperatura. Algunas funciones opcionales como la lectura del puerto serie se han omitido en el diagrama para facilitar su lectura.

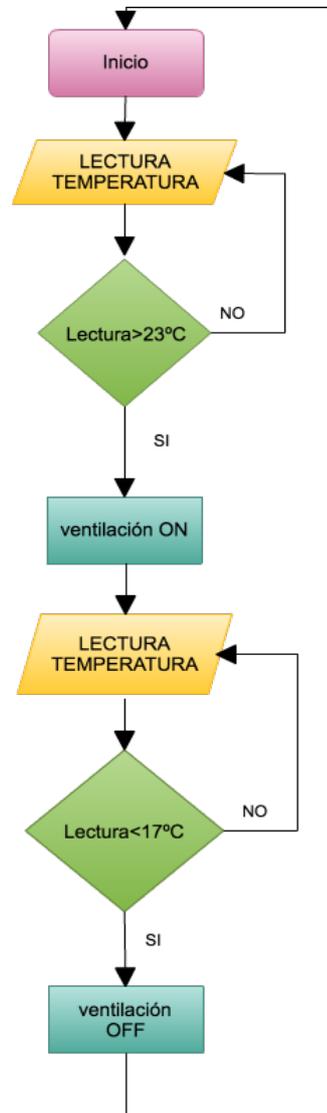


Figura 3-61 Diagrama de flujo del sistema de climatización de la plataforma de impresión

En primer lugar se comienza verificando la temperatura de la caja mediante el sensor de temperatura. Es evidente que la temperatura jamás va a ser de manera natural inferior a 15°C al comenzar la impresión, así que solamente se verifica si la temperatura es superior a 23°C, temperatura que se ha tomado como máxima para la operación; de esta manera se gana tiempo de respuesta en el caso de que la temperatura se esté incrementando hacia valores umbral (25°C). En el caso de que se verifique la condición de que la temperatura sea mayor a 23°C, el ventilador entra en funcionamiento para comenzar a desalojar calor y pasar a continuación a verificar el descenso de temperatura; Se comprueba la condición de que la temperatura sea menor que 17°C para, en caso positivo, apagar el ventilador y volver al punto inicial del programa.

3.7.6.4 Montaje en la impresora del sistema de acondicionamiento de temperatura

Como primer paso, una vez realizado el montaje en la placa de prototipado y el desarrollo del *software*, se ha creído conveniente realizar una comprobación de la precisión del sistema y se han obtenido valores de error de 4 décimas de grado Centígrado sobre las medidas de otro termómetro convenientemente calibrado; estos valores de error son sobradamente válidos para el presente proyecto. En cuanto al montaje del sistema en la impresora, hay que tomar en consideración dos aspectos esenciales, la posición del sensor de temperatura y la posición del ventilador, de tal manera que resulte una configuración óptima en ambos casos.

Es prioridad esencial que ninguno de los dos elementos interfiera en la operación de impresión ocupando espacio destinado a esta actividad, de esta manera se ha decidido que los dos elementos ocupen un lugar en la periferia de la caja, dentro de ella y a una altura superior a la máxima que puede ocupar la cama.

Además, para obtener una lectura lo más precisa posible, se orientará el sensor de temperatura hacia el centro de la caja en el lado opuesto al de instalación del ventilador, así el conjunto estará más equilibrado en el sentido de que las medidas no se verán modificadas por la acción directa del ventilador.

Resulta importante considerar que gradientes acusados de temperatura tanto positivos como negativos, producen resultados negativos en el resultado final de la pieza, tales como grietas o roturas. Teniendo en cuenta este aspecto se ha limitado el voltaje de entrada del ventilador a 5V, así funciona con valores de giro moderados y las variaciones de temperatura se producen de forma lenta.

3.7.7 Sistema de acondicionamiento lumínico del entorno de impresión.

3.7.7.1 Motivación del sistema

La plataforma de impresión remota se ha ideado como un sistema semi-autónomo que permita operar una impresora 3D de la forma más independiente posible en una habitación sin supervisión in-situ. Para obtener una confirmación visual del proceso se ha instalado una cámara web y se plantea el problema de que cuando la luz de la habitación no es adecuada, no se obtiene una buena imagen de video. Esto ha motivado la implementación de un sistema que permita resolver este inconveniente

3.7.7.2 Diseño del sistema de acondicionamiento lumínico

Para resolver el problema descrito se ha ideado un sistema de iluminación del entorno de impresión en función de la luz ambiente, de tal manera que cuando la luz se encuentra por debajo de un umbral establecido, una antorcha LED entra en funcionamiento e ilumina la escena para obtener una imagen de cierta calidad que permita la supervisión del proyecto. (Figura 3-62)

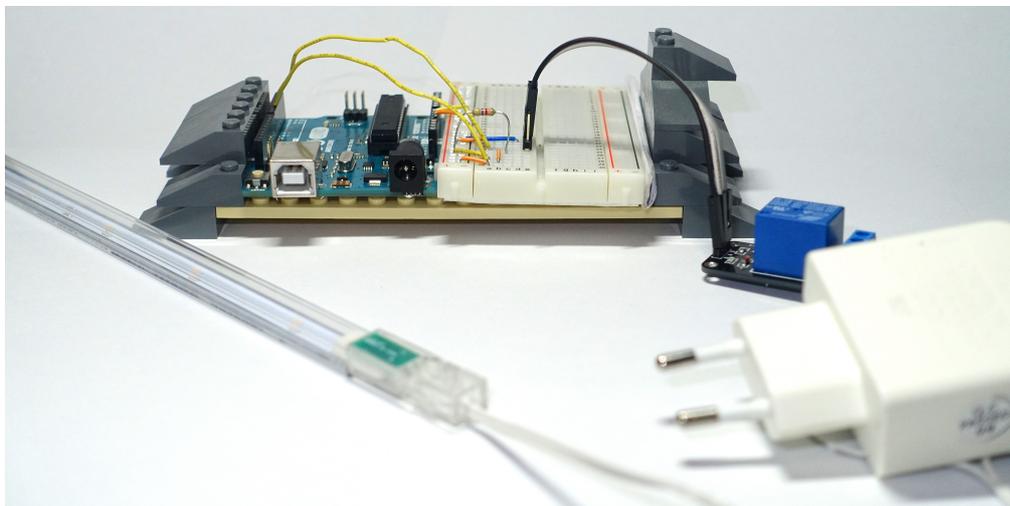


Figura 3-62 Componentes del sistema de acondicionamiento lumínico del entorno de impresión.

El sistema está basado en una placa Arduino que recibe la lectura de la señal analógica de un fotoresistor y en función del valor recibido, se envía una señal digital a un relé que conmuta el circuito que alimenta a la antorcha LED. (Figura 3-63 Esquema del montaje del sistema de acondicionamiento lumínico)

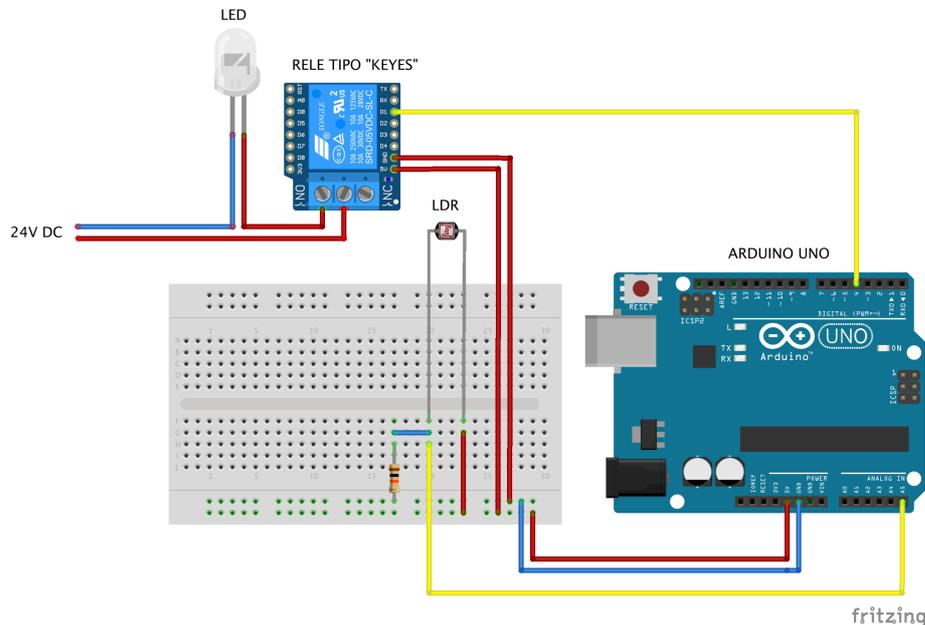


Figura 3-63 Esquema del montaje del sistema de acondicionamiento lumínico

En este sistema la placa Arduino tiene dos funciones principales, gestionar los valores de iluminación de la pieza y conmutar el relé para cerrar el circuito de la antorcha LED. En cuanto a la alimentación del sistema, hay dos fuentes: Por un lado la de 5V que alimenta a la placa (y a su vez al fotoresistor) y por otro lado la fuente de 24V de DC que se consigue a través de un transformador conectado a la red de 220V AC; esto se consigue mandando una señal digital al relé que conmuta el interruptor para cerrar el circuito "NO".

3.7.7.3 Software del sistema de acondicionamiento del sistema de luminosidad

El proceso de programación, al igual que en el descrito en la sección 3.7.7.3, se ha realizado sobre la IDE de Arduino como se muestra en la **¡Error! No se encuentra el origen de la referencia..**

```

Serial.print(suma);
Serial.println(" detector");
delay(100);

if (suma==0){
  ledstate = HIGH;
} else {
  ledstate = LOW;
}

digitalWrite(led, ledstate);
suma=0;
}

void compruebaLuz(){
  luminosidad=analogRead (luz);
  Serial.print(luminosidad);
  Serial.println(" luminosidad");
}

if(Serial.available()) //Nos dice si hay datos dentro del buffer
{
  memset(cadena, 0, sizeof(cadena)); //memset borra el contenido del array "cadena" desde la posición 0 hasta el final sizeof

  while(Serial.available()->0) //Mientras haya datos en el buffer ejecuta la función
  {
    delay(5); //Poner un pequeño delay para mejorar la recepción de datos
    cadena[posicion]=Serial.read(); //Lee un carácter del string "cadena" de la "posicion", luego lee el siguiente carácter con "posicion++"
    posicion++;
  }

  umbral=atoi(cadena); //Convertimos la cadena de caracteres en enteros

```

500
31310 detector
954 luminosidad
23.31 grados Centigrados
31279 detector
955 luminosidad
23.31 grados Centigrados
30721 detector
955 luminosidad
23.31 grados Centigrados
28334 detector
955 luminosidad
23.31 grados Centigrados
21824 detector
955 luminosidad
23.31 grados Centigrados

Figura 3-64 Entorno de programación del sistema de acondicionamiento lumínico

Como funcionalidad a mayores, se ha establecido una conexión mediante el puerto serie con el ordenador; de esta manera Arduino manda al ordenador los valores de luminosidad recibidos cada vez que se ejecuta el bucle y además, desde el ordenador se permite cambiar el valor de umbral de luminosidad si es que es necesario cambiarlo. De esta manera se puede modificar este valor sin ir directamente al código. En el ejemplo de la **¡Error! No se encuentra el origen de la referencia.**, al presionar *enter*, el valor 500 será el nuevo umbral del sistema, modificando el valor por defecto de 650 que se ha establecido para las pruebas en el laboratorio.

En cuanto al funcionamiento del sistema, está basado en un bucle que comprueba continuamente la luz ambiente para así ejercer la acción de encender o apagar la fuente de luz. A continuación se explica de manera gráfica en el siguiente diagrama de flujo (Las funcionalidades opcionales como la comunicación con el puerto serie, se han omitido para facilitar la lectura del diagrama). (Figura 3-65):



Figura 3-65 Diagrama de flujo del acondicionamiento lumínico de la plataforma de impresión

En la Figura 3-65 se puede observar que el programa comienza con una lectura de la luz ambiente con el foto-resistor, a continuación se verifica si dicha lectura está por encima del valor umbral establecido que garantiza una toma de video con unas condiciones de luz correctas.

En el caso de que no sea así, se actúa sobre el relé para encender los LEDs y así iluminar la escena, en el caso de que el valor medido no supere el valor umbral, se siguen tomando lecturas a cada segundo. Una vez la luz está encendida porque la luz ambiente no es suficiente, se continúa con la toma de medidas verificando una a una si la luz ambiente es menor que el umbral, en el caso de que no sea así, se procede al apagado de luz para volver de nuevo al inicio del bucle. En caso contrario se continúa con la toma y verificación de medidas.

3.7.7.4 Montaje del sistema en la impresora 3D

En lo que concierne al montaje, es preciso tener en cuenta dos factores principalmente, la posición del foto-resistor y la posición de la antorcha LED. En cuanto al primer factor, es necesario posicionarlo de manera que capte de manera fiel la luz ambiente y la lectura del mismo no se vea alterada por el encendido de la antorcha LED; este factor puede ser comprobado mediante la visualización de las lecturas en el puerto serie como se explica en la sección 3.7.7.3. El segundo factor tiene que ver con la manera de iluminar la escena, de manera que no queden sombras o "zonas ciegas" que no sean iluminadas.

Por último, es necesario configurar el valor umbral inicialmente para cada una de las habitaciones en las que se fuera a instalar la impresora 3D. De esta manera a través de la ventana de la Figura 3-66, se establece el valor mínimo de luminosidad para el cual se enciende la luz para iluminar la plataforma de impresión.

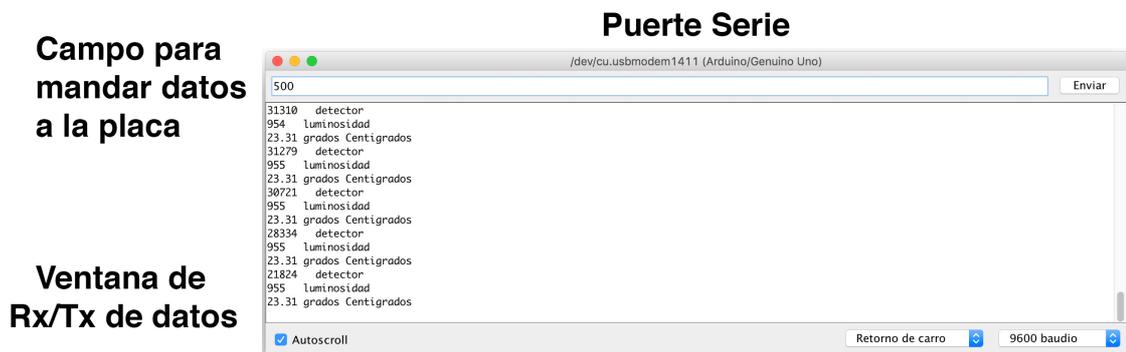


Figura 3-66 Detalle del puerto serie de la IDE de Arduino

3.7.8 Detector de falta de filamento en la impresora 3D

En los siguientes apartados se van a desarrollar los pasos que se han seguido a la hora del desarrollo del sensor de falta de filamento para una impresora 3D. Desde su motivación hasta su implantación en la plataforma.

3.7.8.1 Motivación del detector de filamento

Existe una debilidad común en la gran mayoría de impresoras 3D domésticas del tipo FDM que se pueden encontrar en el mercado, no existe un sistema de avisos que advierta al usuario de que el filamento está cerca de terminarse o un automatismo que pause la impresión si se detecta un bajo nivel de filamento en la máquina. Motivado por estas cuestiones se ha desarrollado un detector de falta de filamento basado en un dispositivo mecánico integrado con Arduino.

3.7.8.2 Diseño del detector de filamento

El sistema combina un accionamiento mecánico con un sistema electrónico sobre Arduino. El detector, contenido en una caja impresa en 3D, basa su funcionamiento en un resorte que se mantiene retraído mientras existe filamento. En el momento que deja de haber filamento, indicando que el nivel remanente es bajo, el resorte se libera y se cierra un circuito. Mediante la lectura de un contacto en una entrada analógica de Arduino, se detecta que el resorte se ha liberado y se actúa sobre la placa Raspberry Pi, cortando la alimentación. De esta manera se pierde la conexión en la impresora 3D y Octoprint y el sistema automáticamente pausa la impresión. Al dejar de existir el servidor web, el usuario pierde la conexión con la impresora permitiéndole conocer que existe algún tipo de error y es necesario supervisar la impresión in-situ.

A la hora de abordar el diseño del sistema, se plantearon algunos objetivos básicos que debía cumplir el detector:

1. Dimensiones reducidas del conjunto.
2. Facilidad de instalación en la impresora 3D sin interferir en la operación normal de la máquina en ningún momento.
3. Independencia total de la impresora, de manera que no exista dependencia en ninguno de los dos sentidos por parte del detector y de la impresora 3D.
4. Diseño que permita ser impreso en la impresora 3D que se dispone.
5. Probabilidad de falso positivo muy baja o nula, de esta manera no habrá interrupciones en la impresión sin motivo.
6. sencillez de uso y mantenimiento muy bajo o nulo.

Teniendo en cuenta todos estos objetivos se llegó a un diseño inicial sobre el software de CAD *Siemens NX v11.0* como se observa en la Figura 3-67.

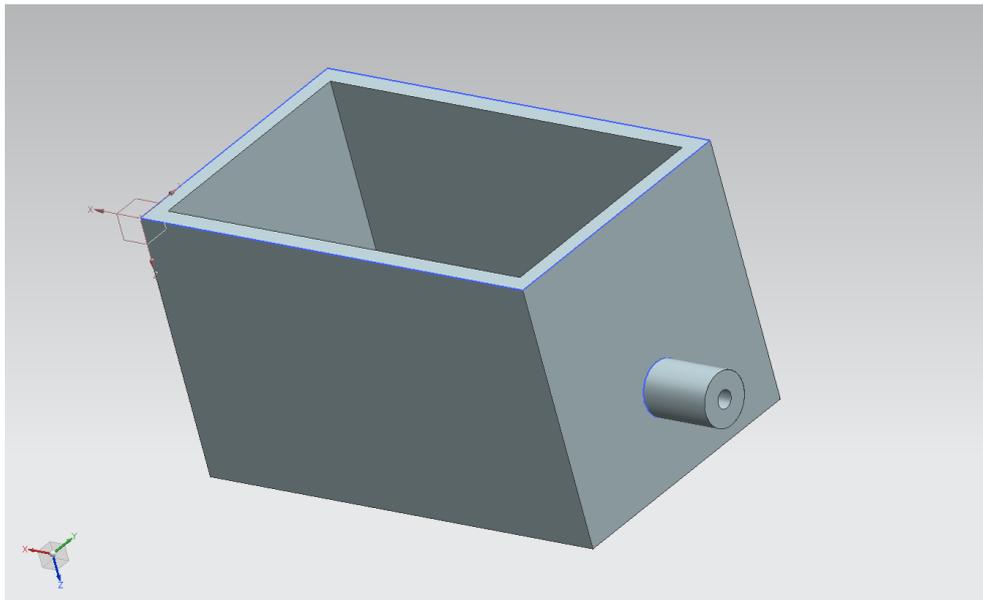
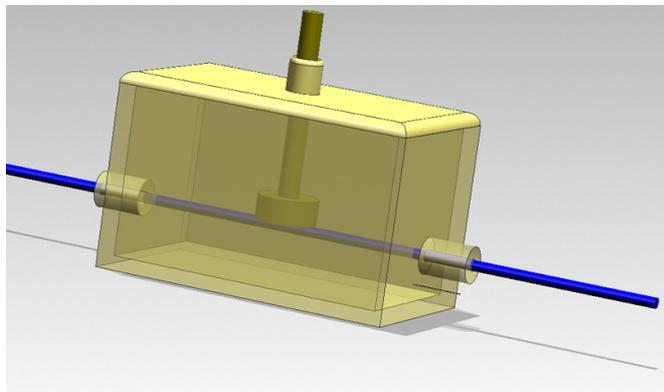
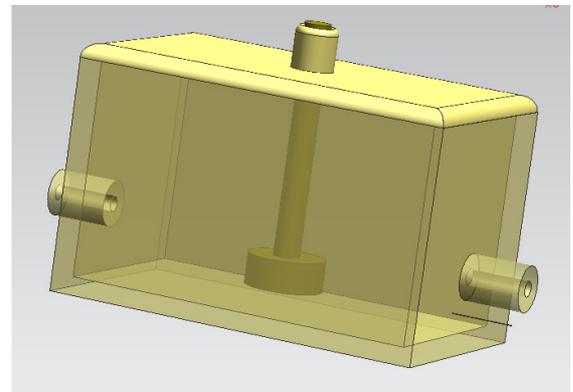


Figura 3-67 Diseño inicial del detector de filamento

El sistema cuenta con 4 piezas: la caja, la tapa de la caja, el eje donde va alojado el resorte y el resorte que permite la retracción del eje; estas tres últimas partes han de ser móviles. Este diseño, en su fase de prototipo, buscaba comprobar en primera instancia la funcionalidad del sistema, verificando su funcionamiento y su baja probabilidad de error. En la Figura 3-68 se puede observar una simulación CAD del funcionamiento del sistema.



Sistema con el resorte retraído debido al filamento (cilindro azul).



Sistema con el resorte liberado debido a que no hay filamento

Figura 3-68 Ensamblaje de diseño funcional del detector de filamento

Una vez verificado el diseño inicial, se procede a su impresión para verificar el funcionamiento en una situación real. Durante el montaje se detectan algunos errores de diseño, como la imposibilidad de soportar en su posición a la tapa de la caja o en el dimensionamiento de los agujeros. Sin embargo, el diseño cumple su función a la hora de detectar la ausencia de filamento. (Figura 3-69, Figura 3-70)

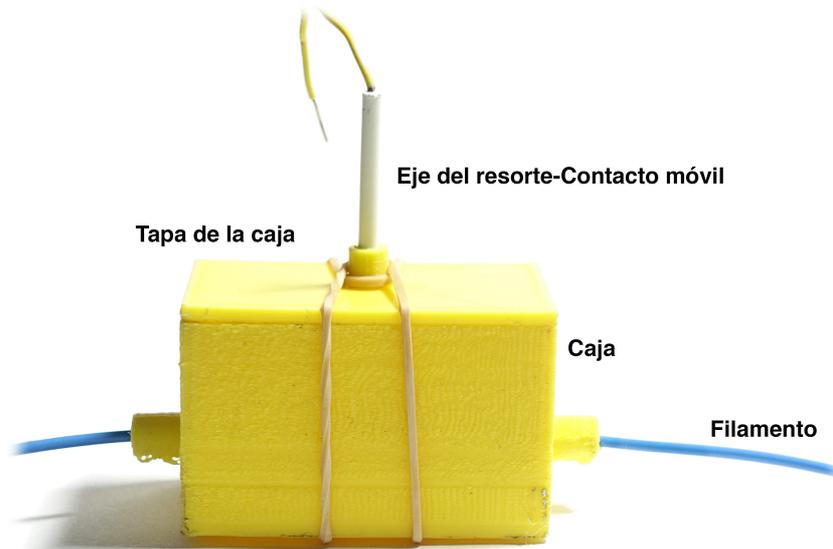


Figura 3-69 Primer prototipo de detector de filamento montado

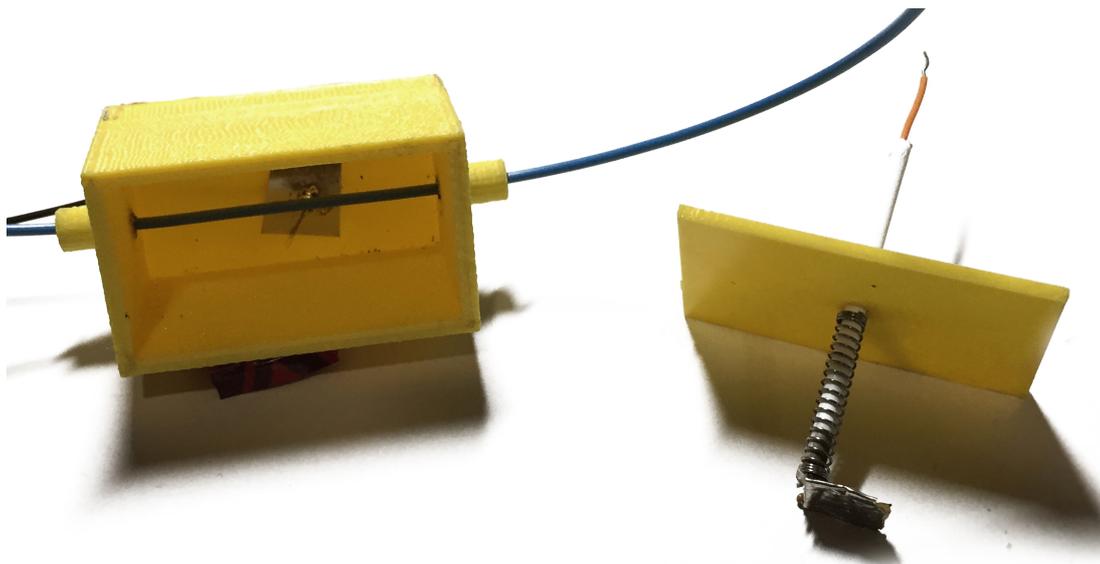


Figura 3-70 Piezas del detector de falta de filamento

Una vez realizado el montaje del sistema, tras haber comprobado el diseño, se pasa a la siguiente fase, un diseño avanzado mejorado más funcional que permitiese corregir los fallos mostrados en el prototipo inicial. (Figura 3-71)

Algunos de los fallos o mejoras que se corrigen son:

1. Miniaturización del conjunto eliminando volúmenes improductivos.
2. Corrección del diámetro del agujero por donde ha de pasar el filamento.
3. Sistema de cierre para la tapa de la caja.
4. Embellecimiento exterior de la pieza.
5. Añadido de una pestaña para su instalación en la bobina de filamento.

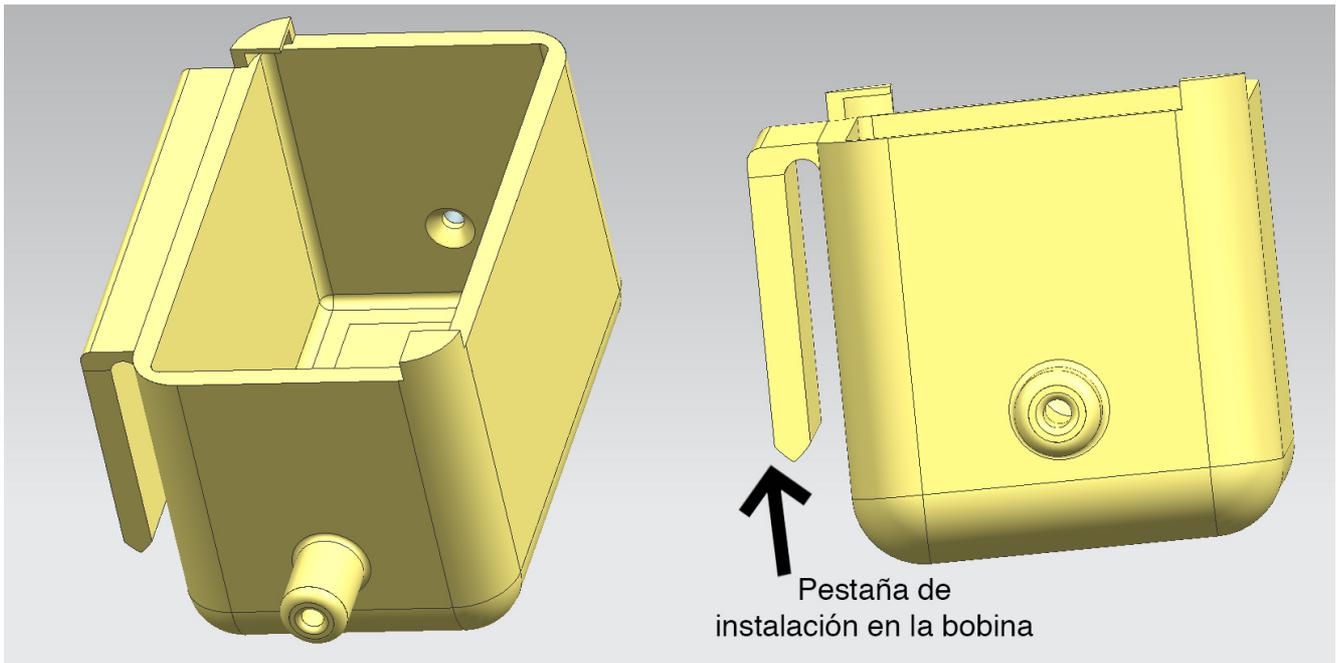


Figura 3-71 Diseño final del detector de filamento

En lo relativo al problema surgido con la sujeción de la tapa, se ha llegado a una solución incluyendo dos pestañas que permiten mantenerla en su posición con un leve giro de la tapa. En la Figura 3-72 se detallan las pestañas instaladas.

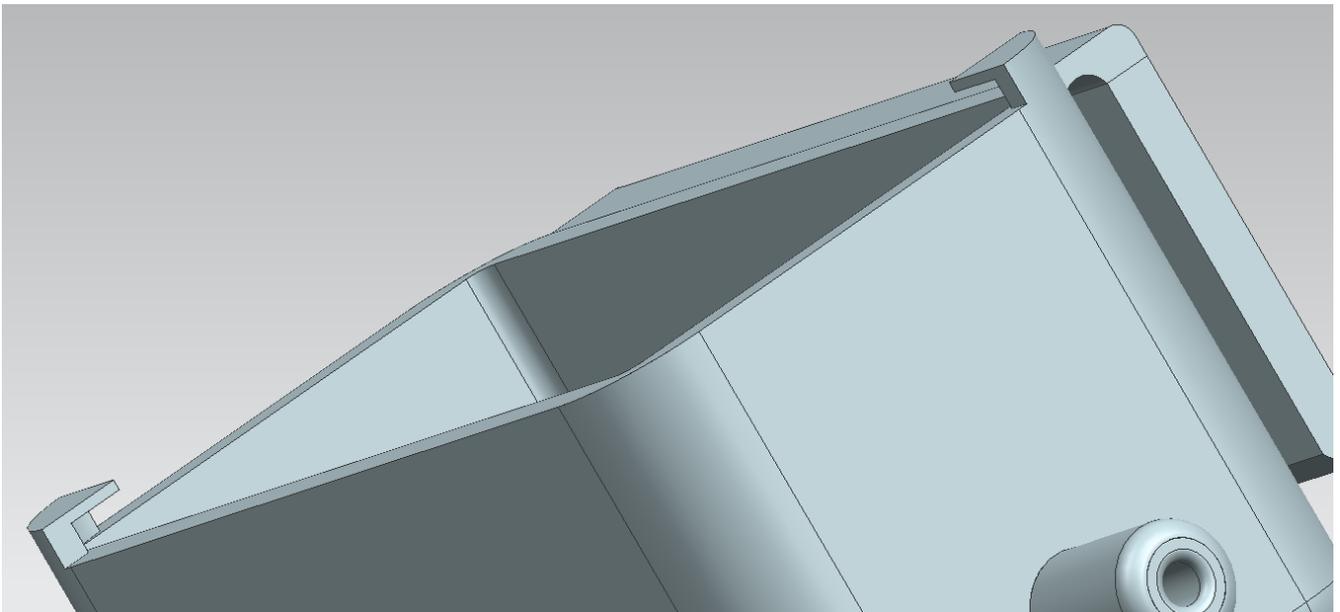


Figura 3-72 Detalle de las pestañas para la sujeción de la tapa

En cuanto al diseño del circuito en Arduino, en un primer momento se piensa en alimentar el contacto retraído con 5V y conectar el contacto inferior a una entrada digital de Arduino, de esta manera, al cerrarse el circuito cuando no hubiera filamento, Arduino recibiría un "1". En la práctica esto no resultó así debido a que el ruido electromagnético creaba falsos positivos en la entrada digital, así que se pensó otra solución.

Finalmente la solución adoptada se basa en conectar el contacto retraído a tierra y el contacto inferior a una entrada analógica; así, la entrada siempre recibe voltaje excepto cuando los dos contactos se unen, que recibe un "0", de esta manera se obtiene la confirmación de que ha dejado de pasar filamento por el detector.

3.7.8.3 Software del detector de filamento

En el siguiente diagrama de flujo se expone el algoritmo del sistema:

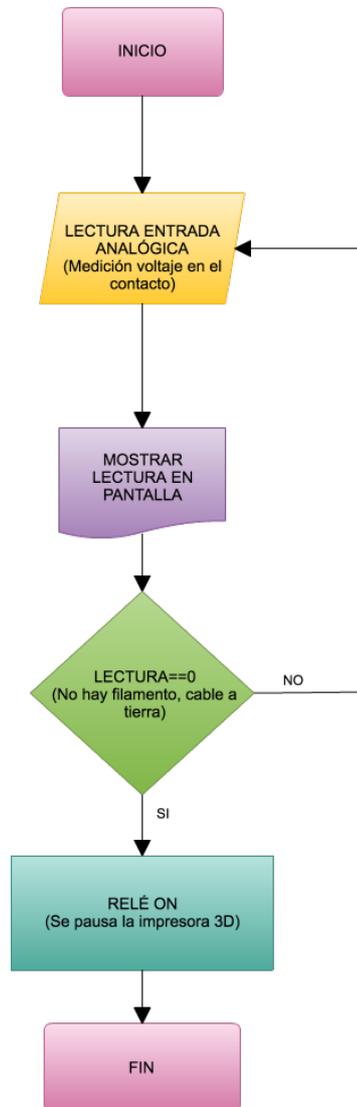


Figura 3-73 Diagrama de flujo del detector de falta de filamento

El proceso comienza con lecturas de la entrada analógica, un número determinado de lecturas se almacena en un *string* y se calcula la media de estas medidas mediante la siguiente fórmula:

$$\frac{\sum Nlecturas}{N} = \overline{Valor}$$

La variable "*Valor*" calculada se muestra en pantalla (en caso de estar conectado el puerto serie) y se verifica si es igual a cero, lo que significaría que ambos contactos están unidos y no hay filamento en la impresor. Es en este punto cuando se conmuta el relé a su estado de "NO" para cortar la alimentación de la Raspberry Pi para cancelar la impresión. Dado que a partir de este punto es condición indispensable la supervisión humana, el programa finaliza.

Es una consideración importante que el hecho de almacenar los valores de las lecturas en un *string* y analizar el valor medio de estos elimina los falsos positivos que pudieran existir por los valores aislados que se registran. El código comentado del programa se adjunta en el Anexo II.

3.7.8.4 Mejora del software del detector de filamento

En base a un *plugin* de Octoprint "Octoprint Filament Sensor" [35] se mejora el sistema de manera que una señal digital de Arduino actúe sobre la placa Raspberry Pi, de esta manera se consigue pausar la impresión sin actuar sobre la alimentación del sistema. Aunque este *plugin* está optimizado para un sensor óptico, se utiliza Arduino para alimentar a un puerto GPIO de la Raspberry Pi con una señal digital que indique "0" (con filamento) o "1" (sin filamento). A continuación se muestra el montaje del *hardware*:(Figura 3-74)

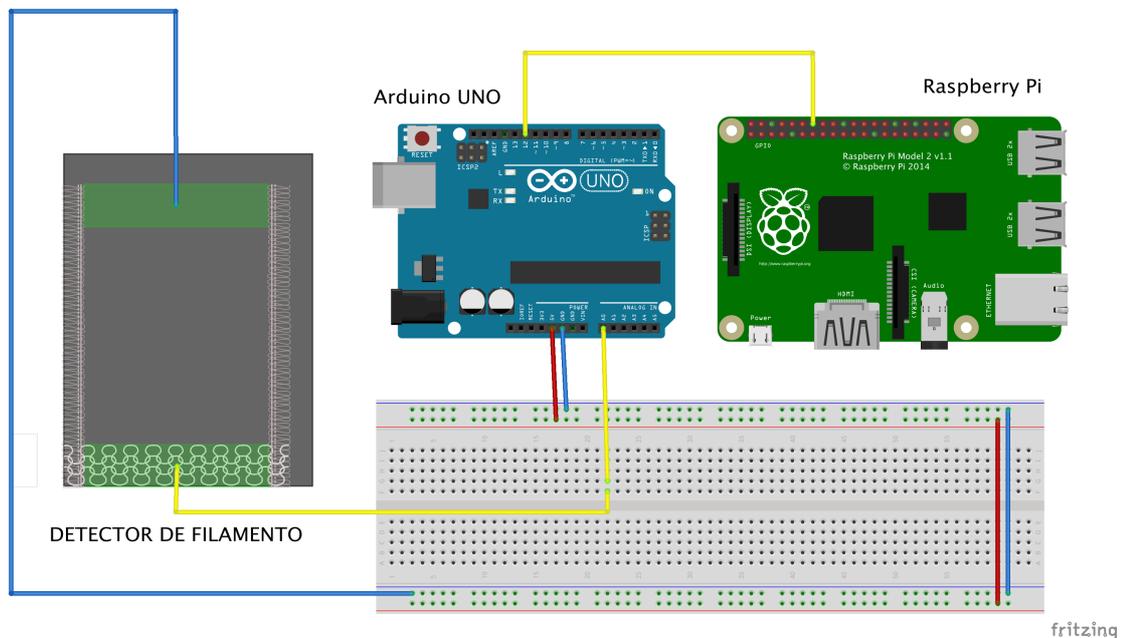


Figura 3-74 Esquema del montaje definitivo del detector de filamento de la impresora 3D

En cuanto al software, se instala en Octoprint el *plugin* descrito en este apartado siguiendo las instrucciones del apartado Configuración de la interfaz web de Octoprint y se procede a su configuración. En primer lugar, se debe añadir al archivo *config.yaml* los siguientes comandos:

```
plugins:
  filament:
    pin: 27
    bounce: 400
```

De esta manera, se lee en el pin GPIO 27 la lectura de la salida digital de Arduino y Octoprint actúa sobre la impresora en función de la señal recibida.

3.7.8.5 Montaje en la impresora del detector de filamento

Para su montaje se han de tener en cuenta ciertos factores:

- Se debe permitir el flujo de filamento sin codos o curvas en el recorrido
- No debe interferir en el funcionamiento correcto de la impresora
- La situación del detector debe estar en una zona en la que se garantice que no va a haber ningún tipo de vibración o movimiento para disminuir al mínimo la probabilidad de falso positivo
- Debe ser un lugar accesible para poder manipularlo y comprobarlo cada vez que comience una operación de impresión.

3.7.9 Software general del sistema

Por último y para cargar todos los programas en Arduino, se ha creado un programa principal en el que cada uno de los sub-programas descritos en los títulos anteriores se ha implementado como una

función que se ejecuta en el *loop*. De esta manera, cada una de las funciones se ejecuta de manera secuencial en bucle, con un *delay* o retraso total de unos 4 segundos para cada vez que se ejecuta el bucle. Este retraso, para el objeto de este programa no supone algo significativo.

La estructura final del código es la siguiente:

1. Declaración de variables
2. *Setup*
3. *Loop*
4. Función Detector de filamento
5. Función Acondicionamiento lumínico
6. Función de climatización

3.7.10 Detector de sobre-intensidades en los motores de la impresora 3D

Con el objeto de proteger la impresora de sobre-intensidad en los motores, se plantea dotar al sistema auxiliar de un sistema de medición de la caída de voltaje en los motores, de esta manera, se podría detectar cuando un motor se atasca para apagar la impresora antes de causar una avería mayor.

En un primer momento, se procede a analizar la placa controladora para estudiar sus posibilidades. Al no existir una señal de salida que proporcione al usuario alguna lectura de los motores, se analizan los *drivers* de los motores paso-a-paso. (Figura 3-75)

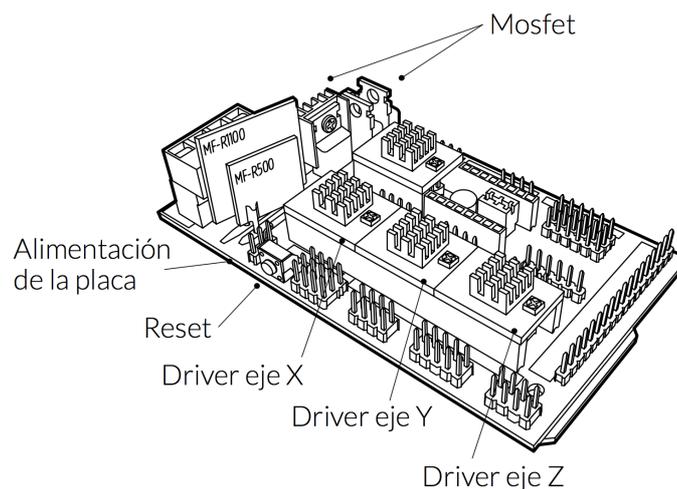


Figura 3-75 Esquema de la placa controladora de la impresora 3D

Una vez localizados los *drivers*, se toman referencias y se realiza una búsqueda del modelo, así como su *datasheet* u hoja de características. Dado que, como ya se ha mencionado, no existe una salida de lectura de los valores de funcionamiento del motor, se piensa en un primer momento en acceder a esta lectura leyendo la caída de tensión en alguna de las resistencias que van a tierra, pero se descarta esta opción debido al reducido tamaño del circuito integrado, ya que esto podría dañar fácilmente la pieza. A continuación se muestra el driver en cuestión instalado en la placa de la impresora 3D (Figura 3-76).

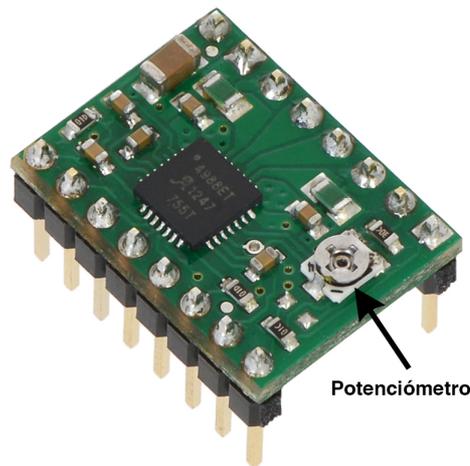


Figura 3-76 Driver A4988 de control de los motores paso a paso

Llegados a este punto, es momento de buscar alternativas que no modifiquen el sistema y permitan garantizar la seguridad en el aspecto descrito. Tras continuar con la investigación, se trata otro de los sistemas complejos de la placa: el *chopping*.

3.7.10.1 Control de la intensidad o *chopping*

En algunos casos, este tipo de *drivers* de control incorporan un sistema de regulación de la intensidad mediante un potenciómetro, esta técnica se conoce como *chopping*. Para ilustrar el funcionamiento, se trabajarán con los valores de los motores paso a paso de la impresora 3D; estos son de $1,5 \Omega$ por fase y una tensión nominal de 12V. En el caso de que se aplique esta tensión, según la Ley de Ohm, pasaría una corriente de 8A por la bobina, lo cual destruiría el motor. Para evitar esto, mediante el limitador de intensidad, se permite utilizar la tensión nominal sin llegar a esos valores tan elevados de intensidad.

Este mecanismo interrumpe la señal mediante una señal digital PWM de forma que el valor promedio de corriente que pasa por la bobina, sea la nominal, el limitador solo dejará pasar corriente un tanto por ciento del tiempo que esté funcionando. El potenciómetro de la placa es capaz de variar este valor umbral de intensidad nominal (Figura 3-76). Para medir esta intensidad umbral basta con medir la caída de voltaje entre el potenciómetro y tierra (GND) y aplicar la siguiente fórmula:

$$I_{max} = 0,625V_{referencia}$$

Esta fórmula es solo una medida aproximación y no supone una medida precisa de intensidad máxima; en la práctica debería realizarse la medida con un amperímetro.

3.7.10.2 Conclusión

Una vez advertida esta medida de seguridad en los drivers de los motores, se llega a la consideración de que la necesidad propuesta está ya cumplida. El único paso necesario que se debe realizar es la correcta configuración del umbral en cada uno de los *drivers*, para dar valores de intensidad máxima más conservadores o permisivos en cada caso. De esta manera se tendrá cubierta la medida seguridad pasiva contra picos de intensidad.

4 RESULTADOS / VALIDACIÓN / PRUEBA

4.1 Primeras pruebas del sistema Octoprint

4.1.1 Conexión entre el ordenador y la Raspberry Pi

Tras la correcta instalación del software, se procede en primera instancia a comprobar la conectividad del sistema. En este aspecto se encuentran los primeros inconvenientes, la causa del error se advierte al hacer una auditoría de seguridad a la red *wi-fi* utilizada en las pruebas del proyecto mediante el comando *nmap*. El puerto 22, el que usa la plataforma Octoprint para establecer la conexión de red, está cerrado.

Este hecho implica un replanteo a la hora de realizar la conexión, finalmente se decide contar con un *router wi-fi* para realizar las pruebas de conexión entre el ordenador y la Raspberry Pi mediante una red LAN. El hecho de no contar con Internet en la red que conecta la impresora con la Raspberry Pi suponía un inconveniente en el apartado de la conectividad del sistema, así que mediante la configuración del *router* inalámbrico como punto de acceso *wi-fi*, se consigue mantener una conexión de área local con acceso a Internet por parte de los dispositivos conectados a esta red. Este hecho supone un aspecto importante en lo relativo a la conectividad. En la Figura 4-1 se muestra la instalación del *router* en el entorno de la impresora.

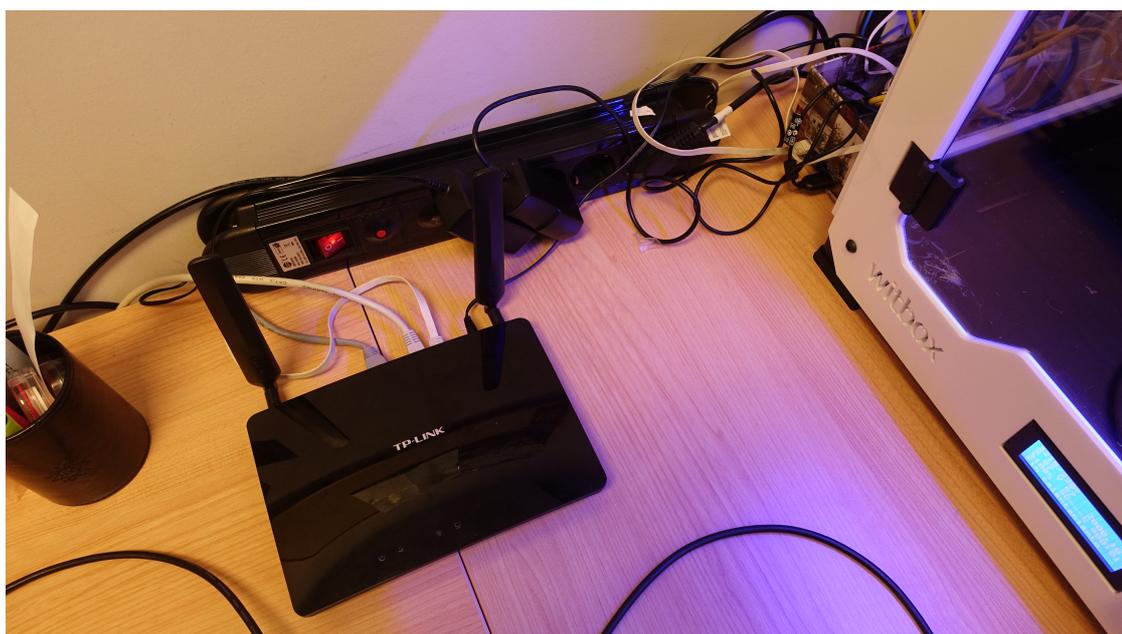


Figura 4-1 Detalle del router inalámbrico en la ubicación actual de la impresora.

4.1.2 Conexión de la impresora a la Raspberry Pi

La impresora, además de poder imprimir piezas en formato *gcode* en su tarjeta SD, sin necesidad de estar conectada a un ordenador, tiene un bus de datos vía USB. Este cable es el utilizado para realizar la conexión impresora-Raspberry Pi. Una vez realizada la conexión física y configurada según los parámetros descritos en el apartado 3.4.3. En la Figura 3-11 se puede observar la interfaz web generada.

4.1.3 Verificaciones en el funcionamiento de Octoprint

En este apartado se revisan las diferentes funciones con las que cuenta el servicio Octoprint, se verifica su funcionamiento y se comentan sus capacidades.

4.1.3.1 Verificación del control de temperaturas

Se comienzan verificando el correcto funcionamiento del ajuste de temperatura del extrusor, ya que la temperatura de la cama no puede ser modificada, la impresora BQ Witbox I no tiene esta capacidad, por lo tanto, este valor permanecerá en OFF. Además se comprueba en la que se grafica correctamente y que mantiene el valor de temperatura objetivo de manera estable. La temperatura óptima para la impresión con PLA se ha establecido en 220° C y efectivamente la leída en la interfaz web (Figura 4-2) coincide con la lectura proporcionada en la pantalla de la impresora.

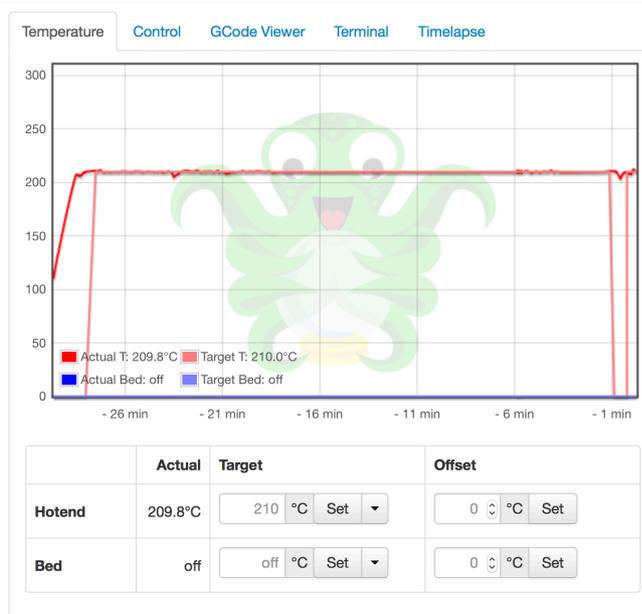


Figura 4-2 Interfaz de ajuste de temperatura del extrusor en Octoprint

4.1.3.2 Verificación del apartado Control de Octoprint

El siguiente paso consiste en verificar el apartado de control de Octoprint. Tras una calibración realizada según el procedimiento explicado en la sección 3.4.3, se procede a manipular los botones y se comprueba que la cabeza del extrusor se puede mover de manera manual, además se comprueba la extrusión y retracción manual del filamento. En lo referente a la operatividad de este apartado, solo cabe mencionar que la impresora no permite apagar el ventilador, con lo cual el actuador **Fans Off** no provoca ninguna respuesta en la impresora.

En este apartado también se permite verificar el *streaming* de la cámara web, configurado para transmitir a 15fps y 240p, resultan unos parámetros suficientes para obtener una buena confirmación visual del proceso de impresión. Sobre su posicionamiento existen algunas consideraciones importantes. La primera prueba que se realiza es con la cámara generando un plano cenital de la impresión sin ningún tipo de soporte (Figura 4-3 Instalación temporal de la cámara web en la

impresora 3D. Durante las primeras pruebas se comprueba que la imagen que se recibe es, en muchas ocasiones, no funcional debido a que el conjunto del extrusor impide la visión de la pieza de manera correcta. Además al realizar trabajos sobre la impresora, la cámara es susceptible de moverse. Motivado por estos hechos se decide diseñar un soporte para la cámara web que permita de manera indistinta realizar un plano por la ventana frontal de la impresora, obteniendo así una vista del alzado de la pieza, o una toma cenital, obteniendo una vista de la planta.

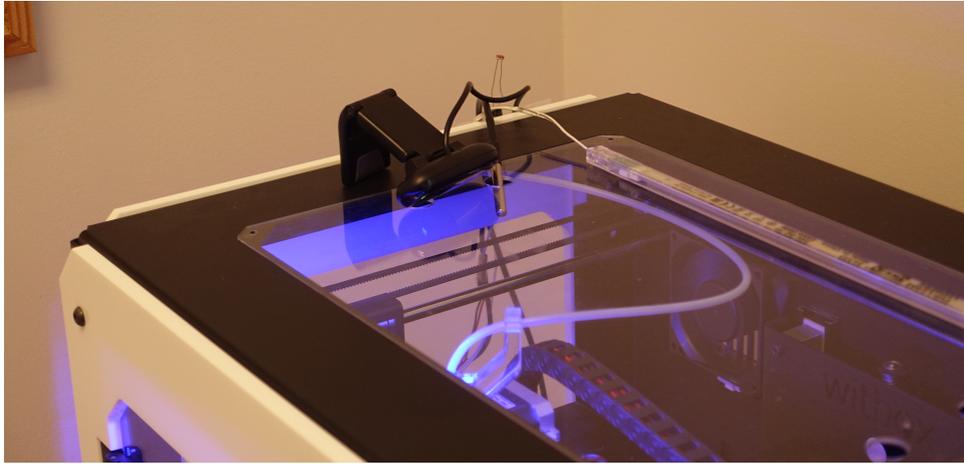


Figura 4-3 Instalación temporal de la cámara web en la impresora 3D

La última función de este apartado es el ajuste de la velocidad de impresión y de la tasa de salida del filamento, sin embargo, se prefiere mantener estos parámetros sin alterar (Figura 4-4) ya que son los perfiles de impresión los que realmente permiten una personalización completa de estos parámetros.

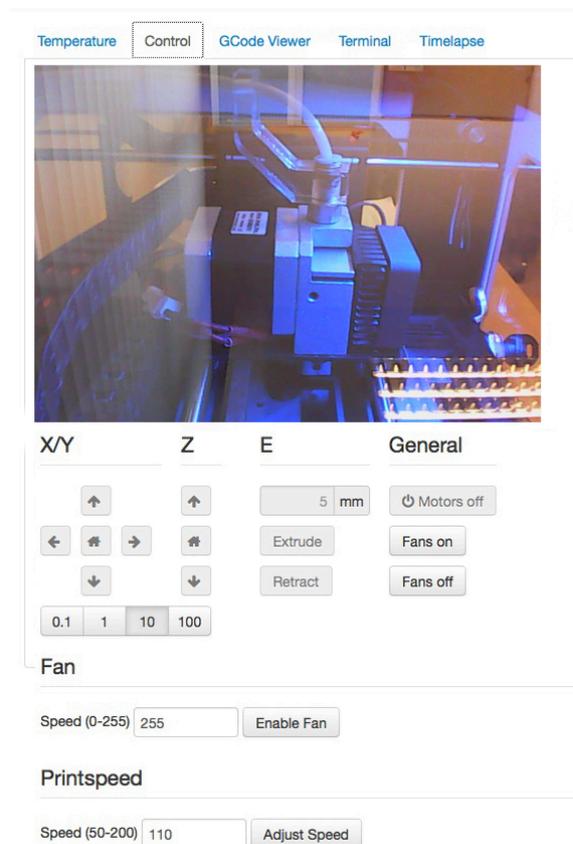


Figura 4-4 Apartado control con *streaming* de una impresión en curso

En cuanto a la validación del apartado **GCode Viewer**, es importante el hecho de que el previsualizador de capas solo genera las imágenes las imágenes del proceso capa a capa si el archivo que se ha cargado para imprimir está en formato *gcode*. En el caso de utilizar un *.stl* esta función quedará desactivada. Como se aprecia en la Figura 4-5, se visualiza de manera correcta la simulación del resultado final de la impresión 3D.

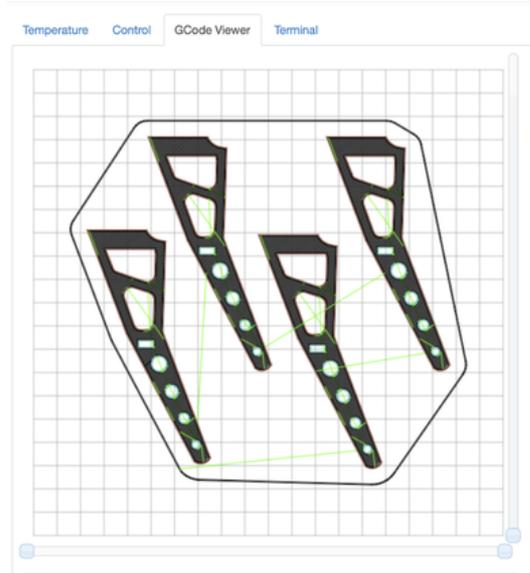


Figura 4-5 Previsualización de fabricación del código *gcode*

4.1.3.3 Verificación del apartado *Timelapse*

Sobre este apartado es interesante una consideración que se advirtió al realizar la primera prueba: La configuración y activación del *timelapse* debe realizarse antes de comenzar a imprimir, una vez el estado de la impresora es el de impresión en curso, este apartado permanece desactivado. En la Figura 4-6 se muestra a modo de ejemplo un fotograma de uno de los *timelapses* realizados.

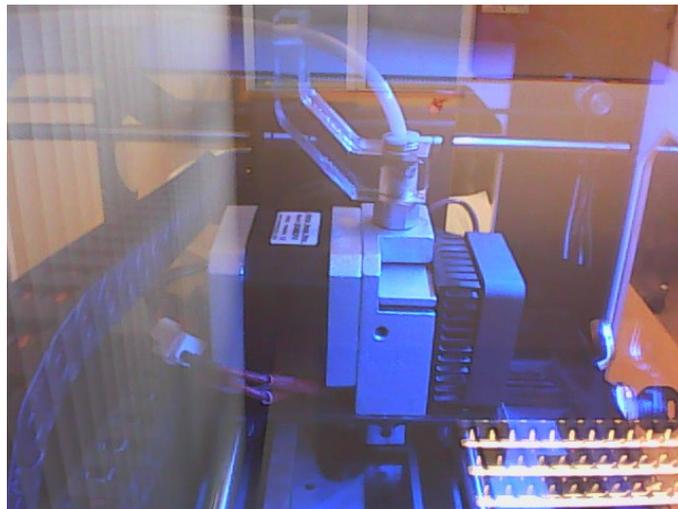


Figura 4-6 Fotograma de uno de los *timelapses* realizados

4.1.4 Carga de los archivos a *Octoprint*

En este aspecto, se verifica que los archivos se cargan de manera correcta con una ligera diferencia dependiendo del formato. En el caso de los archivos *.stl*, *Octoprint* se toma un tiempo para generar las operaciones capa a capa; sin embargo, en el caso de los *gcode*, el archivo, una vez cargado, se encuentra listo para imprimir. Es recomendable, en este aspecto, elegir la opción de cargar el archivo

directamente en la SD de la impresora y no en la Raspberry Pi, de esta manera el proceso de impresión está asegurado contra un posible corte en la conexión.

4.1.5 Primera impresión utilizando Octoprint

Una vez realizadas estas comprobaciones iniciales, se procede a la primera impresión en remoto. Las condiciones de la conexión son a través de una red LAN y en la misma habitación, de manera que se pudiera supervisar el proceso *in situ* en todo momento. En la Figura 4-7 se ilustra el proceso de impresión.

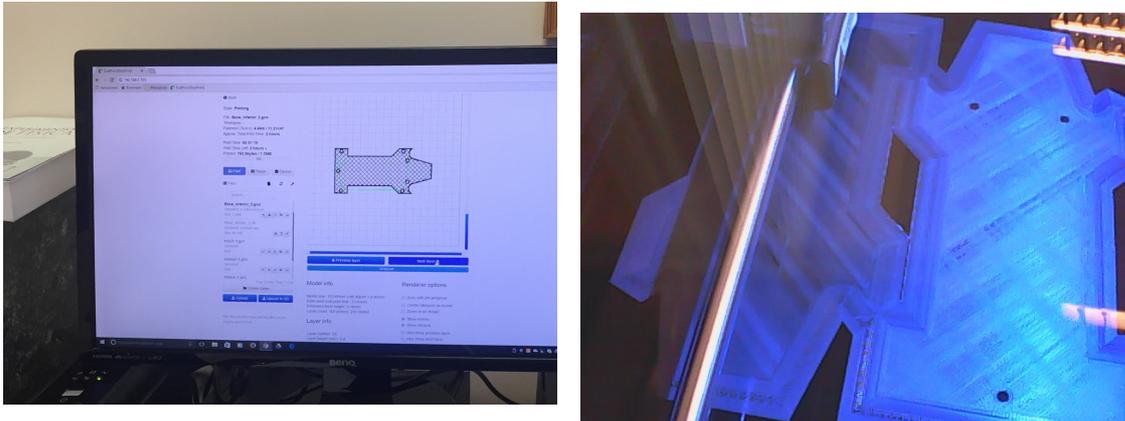


Figura 4-7 Izquierda: Previsualización en Octoprint. Derecha: Producto en impresión

En esta primera impresión se aprovecha el proceso para imprimir dos piezas funcionales del chasis de un dron como parte del Trabajo de Fin de Grado de Carlos Cruz Antón en este mismo Centro. El proceso discurre aparentemente bien y sin problemas, sin embargo, a la concreción del proceso se observa que una de las piezas impresas no está correctamente impresa, existen grietas y hay elementos que no se han concluido.

Con objeto de aislar el problema, se decide repetir la misma impresión de la manera clásica, es decir, sin utilizar el software propuesto. Errores del mismo tipo vuelven a aparecer, indicativo de que la causa de estos errores es, o bien de diseño, o bien del *hardware* de la impresora. Más tarde se descubre que el extrusor de la pieza está obstruido.

4.2 Validación del sistema

Una vez realizada la verificación del software, incluyendo tanto la correcta lectura de los sensores como la realización de acciones de manera acertada, se procede a realizar una impresión completa utilizando Octoprint.

Para la realización de la prueba se continúa con la arquitectura utilizada en la primera prueba, tanto el hardware como el software, además de la topología de red. Una vez más, el proceso, pese a realizarse de manera remota, se realiza en la misma habitación que en la que se encuentra la impresora, de esta manera se permite una monitorización del proceso más en detalle.

4.2.1 Preparación de la pieza

El proceso, si se requiere un control detallado de la impresión, ha de comenzar siempre cargando el diseño CAD en el *software CuraEngine*. Este programa permite tanto orientar la pieza en la impresora. Para el caso de la validación del sistema de impresión, se ha utilizado una geometría que a priori resulta sencilla, pero que requiere de la modificación continua de sus parámetros de impresión. El diseño se trata de una probeta utilizada en las pruebas experimentales del Trabajo de Fin de Grado de Félix Faura García del Toro, con lo cual, en el caso de obtener una pieza impresa correctamente, se

utilizaría en sus pruebas de laboratorio. A continuación se muestra el posicionamiento de la pieza en la simulación de *CuraEngine* (Figura 4-8).



Figura 4-8 Posicionamiento de la probeta plana en la impresora 3D

Una vez verificado este aspecto, se pasan a detallar los parámetros de impresión. Para el caso concreto de esta impresión, se utilizan los parámetros detallados en la Figura 4-9.

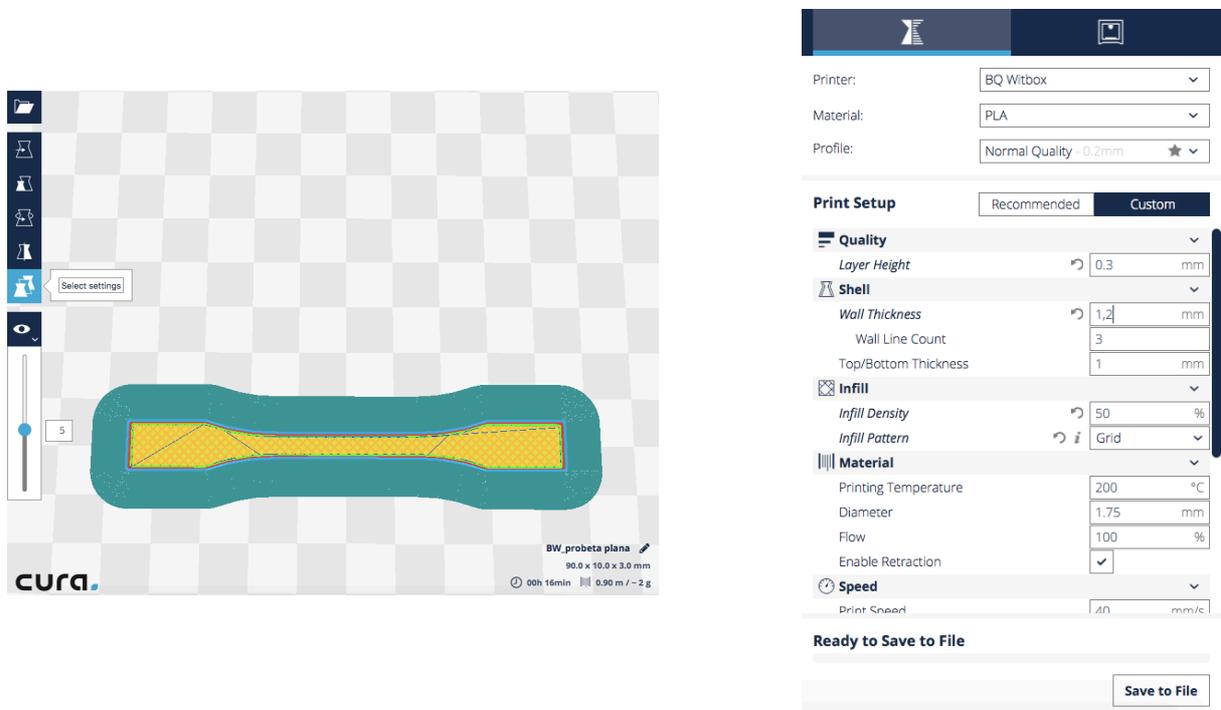


Figura 4-9 Detalle de los parámetros de impresión utilizados

Al acabar de configurar estos parámetros, se guarda el archivo *.stl* y se guarda el perfil de impresión elegido para su posterior exportación a Octoprint como se detalla en el apartado 3.4.3. Una

vez comienza la impresión, se monitorizan los valores de temperatura resultando normales y el proceso discurre sin ningún tipo de inconveniente, pudiéndose adelantar que el sistema cumple con el objetivo principal. La pieza final obtenida mediante el proceso de impresión 3D remota se muestra en la Figura 4-10.



Figura 4-10 Pieza final impresa con una densidad del 50%

4.3 Pruebas y validación del *bot* de Telegram

Tras verificar la necesidad de añadir soporte a la plataforma de impresión remota en dispositivos móviles, se elige Telegram por su sencillez de uso y sus múltiples opciones de conectividad.

4.3.1 Primeras pruebas con el *bot* de Telegram

Estas primeras pruebas se realizan con el objeto de verificar los controles del sistema, la respuesta a los diferentes comandos y ejecutar una evaluación a grandes rasgos de si el valor que se pretende añadir con esta funcionalidad es real y plausible.

Las primeras pruebas se basan en detectar posibles fallos en las respuestas a los comandos enviados. Así pues y a modo de ejemplo, se realizan pruebas de envío de estado forzando a la impresora en diferentes situaciones mediante el comando `/status`. En las siguientes imágenes se puede observar la respuesta del *bot* tanto si la impresora está desconectada de Octoprint como si está imprimiendo. (Figura 4-11)



Figura 4-11 Respuesta del bot al comando /status

Además, se realizan pruebas con los comandos de acción, es decir, los comandos que permiten crear acciones sobre la impresora como comenzar una impresión o modificar los parámetros de la misma. En primer lugar se verifica la acción más básica, conectar la impresora; esto se realiza mediante el comando /con. A continuación, en la Figura 4-12, se muestra la respuesta en la que se pueden observar los diferentes parámetros modificables de la conexión:

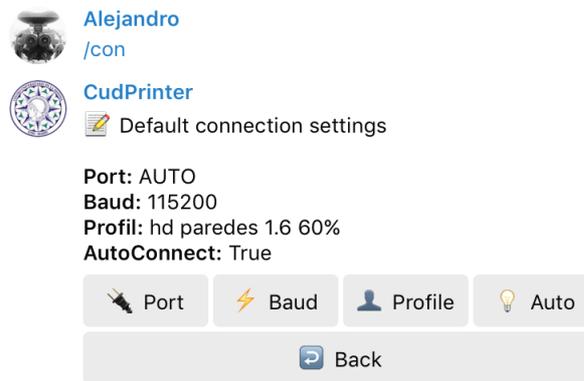


Figura 4-12 Respuesta al comando /con del bot

Otras funcionalidades verificadas son la de carga de piezas a través del dispositivo móvil o comenzar y pausar o cancelar la impresión. (Figura 4-13, Figura 4-14)

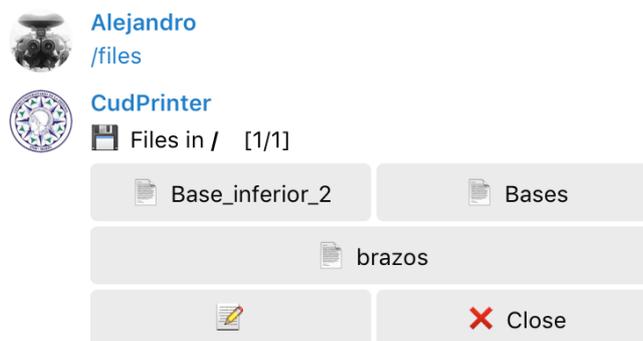


Figura 4-13 Carga y visualización de los archivos disponibles en Octoprint

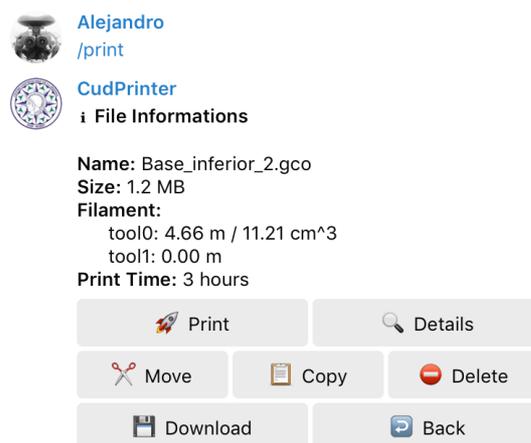


Figura 4-14 Opciones al recibir una respuesta al comando */print*

4.3.2 Validación del sistema.

Si bien por diversos inconvenientes sufridos en la impresora ajenos a este proyecto, no se ha podido realizar una impresión completa utilizando el *bot* de Telegram, si ha resultado ser muy útil en la monitorización y seguimiento de las impresiones en curso. La estimación del tiempo restante de impresión y la confirmación visual han sido una ayuda real para la gestión del tiempo en otros proyectos.

4.4 Pruebas y validación del sistema auxiliar de impresión 3D

En este apartado se analizan, tanto las dificultades en el montaje de este sistema, pasando por los defectos de funcionamiento en relación a este, como las pruebas en lo referente al funcionamiento del software propuesto.

4.4.1 Verificación de la instalación del sistema

Respecto al montaje una vez sobre la impresora, se advierte una ausencia de facilidades para instalar tanto el ventilador, como la antorcha LED y el sensor de luminosidad y temperatura. Siendo así, se decide una colocación provisional simplemente para verificar el correcto funcionamiento del sistema. En la Figura 4-15 se puede observar la colocación de los periféricos para las pruebas.

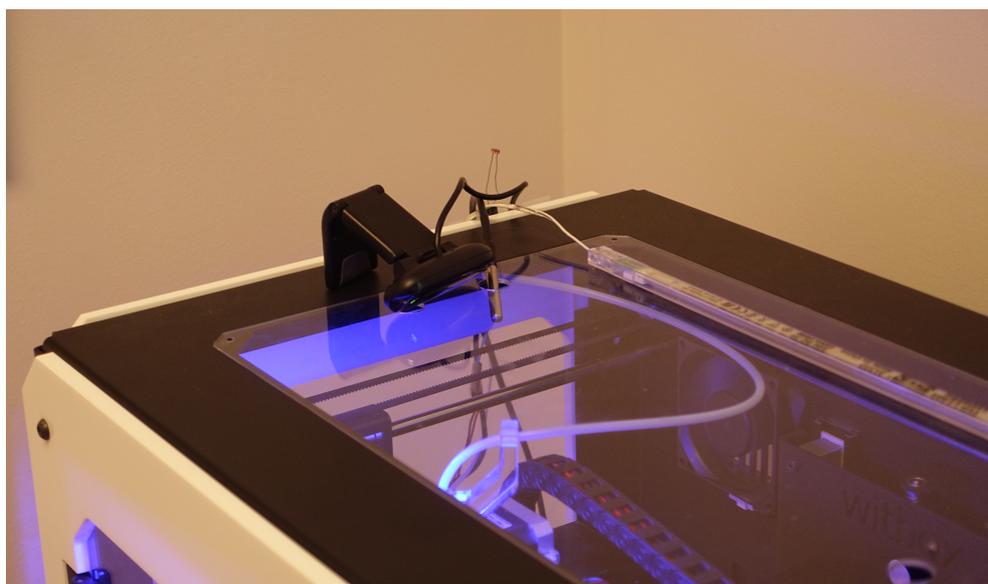


Figura 4-15 Montaje provisional de los periféricos sobre la impresora 3D

Aunque la instalación no es la óptima, el sensor de luminosidad, al ser relativamente direccional, es capaz de detectar lecturas de la luz ambiente desde esa posición. En cuanto al sensor de temperatura, se consigue introducir dentro de la "caja de impresión" y aunque se trata de una posición algo elevada, cabe mencionar que en la gran mayoría de las piezas, la cama se posiciona en la parte más elevada del volumen de impresión, detectándose de esta manera lecturas correctas. Para el caso de la antorcha led, se determina que un montaje óptimo estaría dentro del volumen de impresión, de manera que la luz no tenga que pasar a través de la placa de metacrilato, creando reflejos innecesarios.

4.4.2 Verificación del sistema de acondicionamiento lumínico.

El mayor inconveniente en este sistema es la falta de precisión a la hora de leer la luz ambiente por parte del sensor de luminosidad, ya que con relativa frecuencia se detectan valores que no corresponden con la realidad. Para el caso de la figura, ante un cambio de luz repentino (apagar la lámpara de la habitación) se detectan dos valores que no corresponden con la medida correcta, en primer lugar la primera medida y más tarde, en la transición entre luminoso y oscuro, se recibe también una lectura errónea. En la Figura 4-16 se pueden observar dichas medidas.

```

luz
int luz = A5;
int luminosidad = 0;
int rele = 4;
int relestate = LOW;

void setup() {
  pinMode(rele, OUTPUT);
  pinMode(luz, INPUT);
  Serial.begin(9600);
}

void loop() {
  luminosidad = analogRead(luz);
  Serial.println(luminosidad);
  delay(500);
  if (luminosidad > 650) {
    relestate = HIGH;
  } else {
    relestate = LOW;
  }
  digitalWrite(rele, relestate);
  delay(100);
}

```

Serial Monitor Output:

```

916
953
954
955
955
955
122
303
325
324
323
325
323

```

Figura 4-16 Lecturas leídas por el sensor durante las pruebas del sistema

Tras unas primeras pruebas, aunque estas medidas pueden alterar en cierta medida el funcionamiento del sistema, son solo errores locales, lo cual no supone un error real en el funcionamiento. En cuanto al aporte lumínico de la antorcha LED, las pruebas se han realizado con un conjunto de seis LED's de alta intensidad de 1W y realmente no se ha notado una mejora en la luminosidad si bien si que se advierte un cambio en la temperatura de color del conjunto, convirtiendo, como se puede ver en la Figura 4-17, la luz resultante, en una luz más cálida, dando mejor aspecto a las grabaciones.

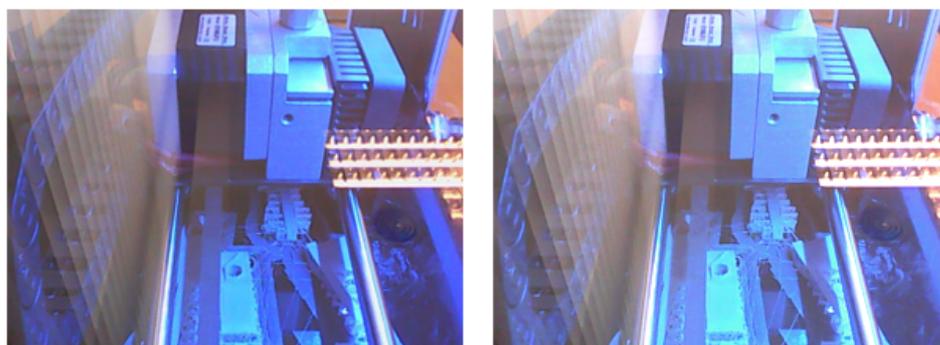


Imagen sin iluminación externa

Imagen con iluminación externa

Figura 4-17 Comparativa de la imagen recibida con luz auxiliar

4.4.3 Verificación del detector de falta de filamento

Existen dos factores principales a considerar en este sistema, en primer lugar el aspecto mecánico, el cual implica principalmente a la liberación del resorte cuando no haya filamento. En segundo lugar, en el aspecto electrónico, se ha de verificar que la entrada al puerto GPIO de la Raspberry produce el efecto deseado, es decir, la puesta en pausa de la impresión.

En el apartado mecánico, en las pruebas realizadas con filamento se producen resultados positivos, aunque durante el desarrollo se realizan mejoras para asegurar esta efectividad al máximo. En el diseño inicial la instalación puede suponer un problema debido a que no existe ninguna "pestaña" para su instalación rápida. La pieza final, que supone la parte crítica del sistema, es la siguiente: (Figura 4-18).



Figura 4-18 Pieza final del contacto del resorte

En cuanto al apartado electrónico, desarrollado en base al *plugin "Octoprint filament sensor"* [35], ha funcionado sin ningún tipo de inconveniente en las pruebas realizadas. Una vez se pausa la impresión por la detección de la falta de filamento, es necesario actuar sobre la impresora manualmente para recargar el filamento en la máquina.

4.4.4 Verificación de sistema de acondicionamiento de la impresora.

En este apartado, pese a que la lectura de la temperatura se recibe con precisión y sin fallos y el software desarrollado funciona (Figura 4-19), encendiendo el ventilador cuando los valores están fuera de los umbrales, debido a la imposibilidad de modificar alguna de las caras de la impresora para instalar el ventilador, no se han podido realizar pruebas *in-situ* de la climatización de la impresora 3D. Sin embargo, se ha propuesto realizar un diseño con capacidad de ser impreso en 3D, en el que se añade espacio para la instalación del ventilador.

```

Temperatura
#include <OneWire.h> //Se importan las librerías
#include <DallasTemperature.h>

#define Pin 5 //Se declara el pin donde se conectará la DATA

OneWire ourWire(Pin); //Se establece el pin de datos

DallasTemperature sensors(&ourWire);

int fan=2;
int estado = LOW;
float temp = 0;

void setup() {
  delay(1000);
  Serial.begin(9600);
  sensors.begin(); //Se inician los sensores
  pinMode(fan, OUTPUT);
}

void loop() {
  sensors.requestTemperatures(); //Se lee e imprime la temperatura en grados Celsius
  Serial.println(" grados Centigrados");

  delay(1000); //Se provoca un lapso de 1 segundo antes de la próxima lectura
  if ( temp > 24 ){
    estado = HIGH;
  } else if ( temp < 17 ){

```

Figura 4-19 Lectura del sensor de temperatura en Arduino

5 CONCLUSIONES Y LÍNEAS FUTURAS

5.1 Conclusiones del presente proyecto

Una vez finalizado el proyecto, es momento de revisar los objetivos propuestos inicialmente para el proyecto y ver cuales se han cumplido y cuales se han quedado pendientes de análisis o revisión.

Las pruebas llevadas a cabo en el apartado 4, en especial aquellas que han testado la capacidad total del sistema, es decir, la capacidad de imprimir partiendo desde un archivo CAD y únicamente la plataforma de impresión remota propuesta, no solo han resultado dados positivos si no que ha resultado un éxito. Se puede afirmar que una plataforma de impresión 3D remota basada en una Raspberry Pi y Octoprint es una opción real para cualquier usuario que precise de esta capacidad.

El sistema diseñado inicialmente no solo proporciona la capacidad de imprimir en 3D de manera remota, sino que tras un análisis de las necesidades de la plataforma, para una mayor capacidad autónoma, se han integrado medios de monitorización tales como una cámara web, o un sensor de falta de filamento que puede pausar la impresión llegado el caso. Además, en la línea de la búsqueda de la operación remota total, se han incorporado diversos sensores tales como un sensor de temperatura o luminosidad controlado mediante Arduino, con el objeto de conseguir que las condiciones de operación sean óptimas en la plataforma de impresión. Además, el sistema descrito procesa toda esta información y toma decisiones básicas, además de compartirla con el usuario operador de la plataforma. Por último, la plataforma desarrollada cuenta con interfaces sencillas e intuitivas con el objeto de que el futuro operador de la plataforma no necesite tener un conocimiento profundo de la misma. De esta manera, cualquier usuario con unos conocimientos básicos podrá acceder a la impresión 3D sin estar directamente en el lugar de la impresora, algo que sin duda añade valor en diversos ámbitos como, en nuestro caso, en educación y Defensa. En la Figura 4-15 se pueden observar los sensores instalados en la impresora 3D.

Como parte de la plataforma se ha implementado una integración con la app de mensajería móvil Telegram, de esta manera se aumentan las capacidades del sistema al ser capaces de operar la plataforma vía Internet. Además, este sistema permite el envío de las notificaciones que el usuario estime oportunas, tales como notificaciones al terminar la impresión o notificaciones horarias, todas ellas acompañadas de una fotografía en vivo tomada con la cámara web. En la **¡Error! No se encuentra el origen de la referencia.** se puede observar la interfaz de Telegram comunicándose con la impresora 3D.

En cuanto al hardware utilizado, queda patente que mediante sistemas de código abierto y con un coste contenido, se ha podido obtener una plataforma remota de impresión que cumple con los requisitos que se impusieron en el apartado 3.1.1, tales como fiabilidad o estabilidad sin que este factor del coste haya resultado limitante a la hora de cumplir con los objetivos impuestos. Además, el uso de

estos sistemas abre las puertas a futuras implementaciones para conseguir una plataforma personalizada acorde con las necesidades que se busquen en cada caso.

5.2 Opinión personal

La plataforma propuesta representa una alternativa real a la impresión 3D "tradicional" que, sin añadir nuevas capacidades en lo respectivo a la operación de la misma, sí que abre nuevas puertas en cuanto a la conectividad y la monitorización. Proyectos que a priori parecen seguir líneas paralelas como el sistema de impresión remoto y el sistema de monitorización de condicionantes externos, finalmente se han hecho converger, para, de alguna manera, conseguir un conjunto robusto, estable, y fiable, requisitos que resultan fundamentales en el desarrollo de una plataforma de estas características.

Habiendo hecho un estudio de la impresión 3D doméstica actual, sus capacidades y sus puntos débiles, creo firmemente que el trabajo realizado añade un valor real a la plataforma actual de impresión 3D.

5.3 Continuación del presente proyecto

Debido a causas ajenas al proyecto, la impresora 3D ha contado con ciertas averías que han impedido realizar todas las pruebas que hubieran concluido el proyecto completamente. Se estima que al menos dos pruebas completas más sobre la impresora son necesarias para verificar el proyecto al completo. Además, ciertas piezas diseñadas para la integración del sistema auxiliar en el entorno de la impresora no han sido impresas debido a estos problemas. A continuación se enumeran las tareas que sería conveniente realizar en un futuro para dar el proyecto por concluido.

- Realización de dos pruebas más sobre la plataforma de impresión para completar la verificación del sistema.
- Impresión de la caja para Arduino.
- Impresión del brazo para la cámara web y la placa modular para integrar el ventilador
- Realización de pruebas de climatización sobre la impresora durante un proceso de impresión 3D.

5.4 Líneas futuras de estudio

Como se ha mencionado anteriormente, dada la naturaleza de código abierto de los sistemas utilizados, existen multitud de posibles nuevas vías de estudio e investigación para conseguir implementar más capacidades e integrar la plataforma en más servicios. A continuación se van a analizar a grandes rasgos las capacidades que a priori se han determinado más interesantes.

5.4.1 Integración de la plataforma con una aplicación para Android

Siguiendo en la línea de código abierto y para dar más libertad a la plataforma de la que ofrece Telegram, resulta interesante el estudio de una aplicación que permita el control y monitorización de la plataforma de impresión 3D remota. Un *centro de control*, desde el cual se monitoricen todos los sensores y se permita total libertad de acción sobre la impresora 3D.

5.4.2 Integración de un sistema de luz Ultravioleta para sintetizar el material de impresión.

Existen pocos estudios relacionados con la utilización de luz UV en impresión 3D doméstica del tipo FDM para disminuir los tiempos de curado y aumentar su dureza. Un sistema que incorpore automatismos con el uso de este tipo de luz para mejorar las propiedades del proceso de fabricación y del material de impresión en sí podría resultar muy positivo. Además, hoy en día existen LED's que, a un bajo coste, permitirían iniciar investigaciones en este aspecto. [36]

5.4.3 Integración de la plataforma remota de impresión 3D en la red intranet de la Armada.

Uno de los retos relacionados directamente con la capacidad de control remoto de la impresión 3D y el mundo militar, es el de integrar una red de impresión 3D dentro de la red de Defensa. De esta manera, sin un gran coste de infraestructura, sería posible la instalación de impresoras en Unidades a flote y en tierra, integrando en mayor medida a la Armada Española en las Fuerzas Armadas del siglo XXI. En la siguiente imagen (**¡Error! No se encuentra el origen de la referencia.**) se propone una Red Nacional de impresión 3D remota. De esta manera se podría coordinar desde un Centro la gestión de la impresión 3D remota.

6 BIBLIOGRAFÍA

- 1] Thingiverse, «Thingiverse projects,» 12 febrero 2014. [En línea]. Available: <http://www.thingiverse.com/thing:247671>. [Último acceso: 13 febrero 2017].
- 2] T. N. Stinson, «navy,» 6 junio 2014. [En línea]. Available: http://www.navy.mil/submit/display.asp?story_id=81936. [Último acceso: 9 febrero 2017].
- 3] J. E. L. Conde, Notas de futuro. Impresoras 3D, Madrid, Madrid: Centro de Análisis y Prospectiva. Gabinete Técnico de la Guardia Civil, 2016.
- 4] Makerbot, «makerbot,» 2017. [En línea]. Available: <https://www.makerbot.com/>. [Último acceso: 10 febrero 2017].
- 5] NASA, «NASA,» 11 noviembre 2016. [En línea]. Available: https://www.nasa.gov/mission_pages/station/research/experiments/1115.html. [Último acceso: 8 febrero 2017].
- 6] Stratays, «www.stratays.com,» 2015. [En línea]. Available: <http://www.stratays.com/es/industrias/aeroespacial-militar/airbus#sthash.8wQHZwbT.dpuf>.
- 7] Airbus, «airbus,» 2017. [En línea]. Available: <http://www.airbus.com/aircraftfamilies/passengeraircraft/a350xwbfamily/a350-800/>. [Último acceso: 8 febrero 2017].
- 8] L. Kemel, «www.materialise.com,» 27 Julio 2015. [En línea]. Available: <http://www.materialise.com/blog/phoenix-childrens-hospital-runs-3d-printing-project-for-patients/>. [Último acceso: 24 Enero 2017].
- 9] 3D Systems, «www.3dsystems.com,» 2017. [En línea]. Available: <http://www.3dsystems.com/company>. [Último acceso: 8 febrero 2017].
- 10] C. Hull, Apparatus for Production of Three-Dimensional Objects by Stereolithography, U.S Patent 4.575.330, 1986.
- 11] C. Deckard, Method and apparatus for producing parts by selective sintering, USPTO n° 4863538, 1986.
- 12] S. A. Sarmiento, «www.youngmarketing.com,» 4 mayo 2016. [En línea]. Available: <http://www.youngmarketing.co/la-cultura-del-maker-movement-y-como-esta-cambiando-el->

- mundo/. [Último acceso: 16 Enero 2017].
- 13] 3DHubs, «3DHubs,» 2017. [En línea]. Available: www.3dhubs.com/trends. [Último acceso: 14 febrero 2017].
- 14] W. Palm, «Rapid Prototyping Primer,» Penn, 2008.
- 15] technabob, 2017. [En línea]. Available: http://technabob.com/blog/wp-content/uploads/2014/01/3d_chocolate_printer.jpg. [Último acceso: febrero 2017].
- 16] Wobbleworks, «the3doodler,» 2017. [En línea]. Available: <http://the3doodler.com/>. [Último acceso: 5 febrero 2017].
- 17] WobbleWorks, «www.wobbleworks.com,» [En línea]. Available: www.Wobbleworks.com. [Último acceso: 20 Enero 2017].
- 18] J. F. Francolí y R. B. Díaz, Estado actual y perspectivas de la impresión en 3D, vol. 1, Barcelona, Cataluña: Generalitat de Catalunya, Dpto. de Empresa y Empleo, 2014.
- 19] CATO Institute, Technologies Converge and Power Diffuses, 2016.
- 20] Impresión tresde, «<http://impresiontresde.com>,» 2016 mayo 2016. [En línea]. Available: <http://impresiontresde.com/blog/la-impresion-3d-militar-amenaza-y-oportunidad/>. [Último acceso: 25 enero 2017].
- 21] R. J. Pacheco, «Tecnologías de impresión 3D para su uso logístico en operaciones,» *Congreso Nacional de I+D en Seguridad y Defensa*, 2014.
- 22] La catedral del Mar, «www.horizonteantartida.es,» 24 Enero 2017. [En línea]. Available: <http://www.horizonteantartida.es/la-foto-del-dia/impresora-3d-en-la-antartida-anticipando-el-futuro/>. [Último acceso: 25 enero 2017].
- 23] OctoPrint, «www.Octoprint.org,» [En línea]. Available: www.Octoprint.org. [Último acceso: 22 enero 2017].
- 24] Astroprint, «www.astroprint.com,» 2017. [En línea]. Available: <https://www.astroprint.com/>. [Último acceso: 10 febrero 2017].
- 25] Raspberry Pi Foundation, «www.raspberrypi.org,» [En línea]. Available: <https://www.raspberrypi.org/>. [Último acceso: 16 enero 2017].
- 26] "Astro Pi Mission", «Astro Pi Mission,» 2015. [En línea]. Available: astro-pi.org/about/mission. [Último acceso: 24 Enero 2017].
- 27] M. P. Estes, «www.geekytheory.com,» 2016. [En línea]. Available: <https://geekytheory.com/tutorial-raspberry-pi-5-crea-un-transmisor-de-radio-fm/>. [Último acceso: 19 Enero 2017].
- 28] A. Wheeler, «www.3dprintingindustry.com,» 3 octubre 2014. [En línea]. Available: <https://3dprintingindustry.com/news/actually-worlds-first-raspberry-pi-powered-3d-printer-34018/>. [Último acceso: 23 enero 2017].
- 29] Arduino Foundation, «<http://playground.arduino.cc>,» [En línea]. Available: <http://playground.arduino.cc/Learning/OneWire>. [Último acceso: 11 Febrero 2017].

- 30] | Bq, Manual de usuario. Witbox, Madrid: Mundo Reader S.L, 2013.
- 31] | Apple, «www.support.apple.com,» [En línea]. Available: <https://support.apple.com/es-es/bonjour>. [Último acceso: 1 febrero 2017].
- 32] | Imprimalia, «<http://imprimalia3d.com/>,» 26 octubre 2016. [En línea]. Available: <http://imprimalia3d.com/noticias/2016/10/25/008144/leon3d-desarrolla-primer-abs-efecto-warping>. [Último acceso: 12 febrero 2017].
- 33] | Telegram messenger, «Telegram,» 2017. [En línea]. Available: <https://telegram.org/>. [Último acceso: febrero 2017].
- 34] | MakerBlock, «<https://www.makerbot.com>,» 23 6 2011. [En línea]. Available: <https://www.makerbot.com/media-center/2011/06/23/12-ways-to-fight-warping-and-curling>. [Último acceso: 8 2 2017].
- 35] | MoonshineSG, «<https://github.com>,» 2 febrero 2016. [En línea]. Available: <https://github.com/MoonshineSG/OctoPrint-Filament>. [Último acceso: 16 febrero 2017].
- 36] | M. N. Karim, «Towards UV curable of biodegradable poly,» New York, 2015.
- 37] | Microjpm, «microJPM,» 2017. [En línea]. Available: <http://www.microjpm.com/products/relay-srd-05vdc-sl-c-spdt-5vdc/>. [Último acceso: 20 febrero 2017].
- 38] | «Web de La Moncloa,» [En línea]. Available: <http://www.lamoncloa.gob.es>. [Último acceso: 13 enero 2015].
- 39] | J. Rodríguez y V. Fernández, Cómo redactar el estado del arte de un trabajo, Editorial Genios, 2010.
- 40] | P. Martínez y A. García, Cómo escribir una buena memoria de TFG, Publicaciones del 2000, 2013.
- 41] | C. H. G. d. Valle, Video Surveillance System based on Raspberry Pi and Pi Camera, Marín, Pontevedra: Centro Universitario de la Defensa de Marín, 2016.
- 42] | NASA, «www.nasa.gov,» [En línea]. Available: https://www.nasa.gov/mission_pages/station/research/experiments/1115.html#publications. [Último acceso: 2017 enero 20].
- 43] | G. Hodgson, «garyhodgson,» 2016. [En línea]. Available: <http://garyhodgson.com/reprap/wp-content>. [Último acceso: 25 febrero 2017].

ANEXO I: ESPECIFICACIONES TÉCNICAS.

Especificaciones técnicas de la impresora 3D utilizada en el proyecto

- Velocidad de impresión hasta 200 mm/s.
- Temperatura de funcionamiento óptimo entre 15° C y 25° C.
- Resolución de hasta 20 micras.
- Volumen de impresión: (x)297 x (y)210 x (z)200 mm.
- Filamento : PLA, derivados de PLA y Filaflex.
- Fuente de alimentación 12V 150W.
- Peso: 30 kg.
- [30]

Especificaciones técnicas de los componentes de la plataforma de impresión 3D remota

1. Raspberry Pi model B:

- Memoria RAM: 512 mb
- Conexiones: 4 puertos USB, puerto Ethernet, Jack 3,5mm de audio, salida HDMI, salida Composite Video.
- Procesador: Broadcom BCM2835

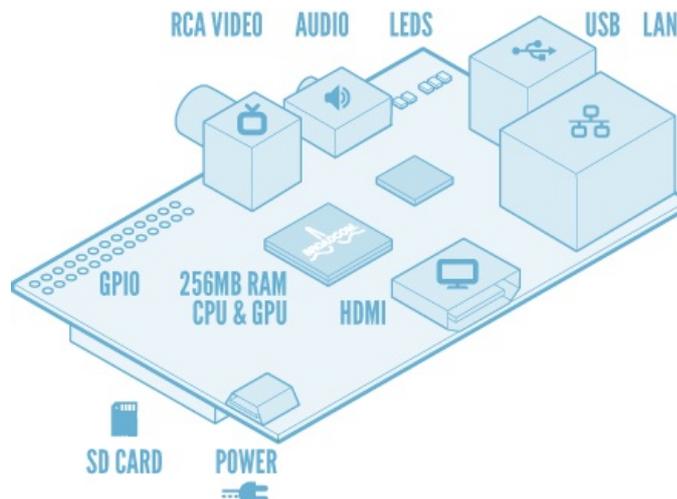


Figura A-1 Esquema de conexiones de la Raspberry Pi

2. Puertos GPIO:

- $V_{in\ GPIO}$: 3,3 V
- Diferentes salidas a 5V y 3,3V (Figura A-2)

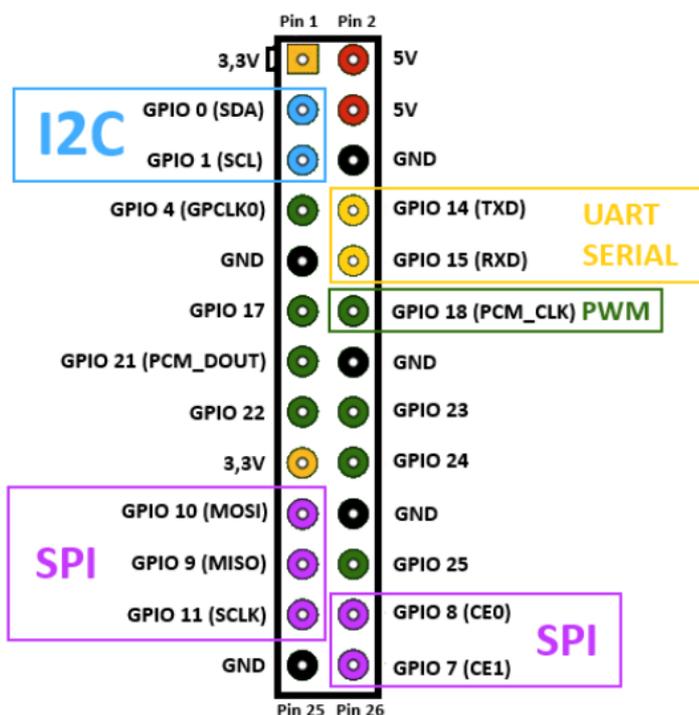


Figura A-2 Esquema de los puertos GPIO de la Raspberry Pi Model B

3. Cámara web Logitech C170

- Resolución máxima de videoconferencia: 640x480 píxeles
- Sensor de 5 megapíxeles
- Conectividad USB 2.0
- Toma de imágenes a 2 megapíxeles
- Resolución máxima de videoconferencia: 640x480 píxeles

4. Router inalámbrico TP LINK MR200

- Interfaz: 3 Puertos LAN 10/100Mbps, 1 Puerto LAN/WAN 10/100Mbps.
- Fuente de alimentación: 12V/1A.
- Estándar inalámbrico: IEEE 802.11ac/n/a 5GHz, IEEE 802.11b/g/n 2.4GHz
- Frecuencia: 2.4GHz y 5GHz.
- DNS Dinámico: DynDNS.

Especificaciones técnicas de los componentes del sistema auxiliar

1. Sensor digital de temperatura DS18B20:

- Medición de temperaturas desde -55°C hasta 125°
- Resolución de 0,5°C en el intervalo (-10°C, 85°C)
- Resolución programable de 9 a 12 bits
- La recepción de datos se realiza mediante el protocolo **1-wire** con Arduino.

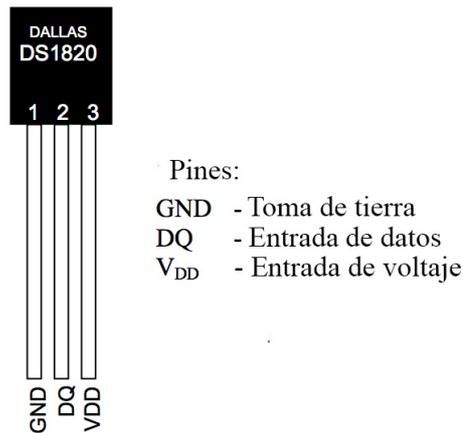


Figura A-3 Esquema de conexión del sensor de temperatura

Antorcha LED:

- V_{in} : 24V DC
- Potencia lumínica: 3W/LED
- 5 LED's por módulo

2. Relé tipo "keyes":

- Especificaciones de la bobina:
 - Voltaje nominal: 5V
 - Corriente nominal: 70 a 90mA
 - Resistencia: 55 a 70 ohm
 - Potencia: 0.45W
- Especificaciones del contacto:
 - Capacidad para carga resistiva: 28V (DC) 7A, 125V (AC) 10A, 240V (AC) 7A
 - Capacidad para carga inductiva: 120VAC 5A, 28VDC 5A
 - Máximo voltaje: 250VAC

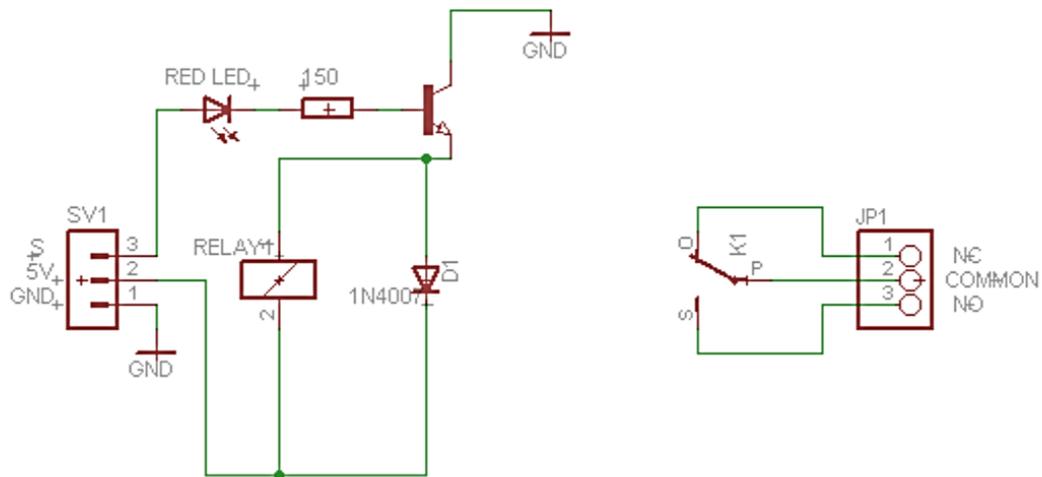


Figura A-4 Esquema del circuito del relé tipo Keyes [37]

3. Ventilador DC

- Velocidad de giro: 1800 rpm a 12V
- Voltaje de entrada variable entre 5 y 12V
- Bajo nivel de ruido con un máximo de 0,4 sonios.

Resistencia LDR

- Velocidad de variación de la resistencia: 50ms
- Valores de funcionamiento: De $1M\Omega$ en oscuridad total a 50Ω .

ANEXO II: CÓDIGO DEL SISTEMA AUXILIAR

```
// -----  
// SISTEMA DETECCIÓN FINAL FILAMENTO  
// -----  
  
int entrada = A0;  
int LECTURA;  
int valores[30];  
int i = 0;  
int suma = 0;  
int led = 13;  
int ledstate = LOW;  
  
void detectorfilamento(){  
    // se almacenan 30 valores de voltaje  
    LECTURA = analogRead(entrada);  
    for (int i= 0; i <= 30; i++){  
        valores[i]= LECTURA;  
        suma= suma + LECTURA;  
        delay(100);  
    }  
    // se imprime la suma  
    Serial.print(suma);  
    Serial.println("  detector");  
    delay(100);  
    // suma == 0, se detecta final de filamento  
    if (suma==0){  
        ledstate = HIGH;  
    } else {
```

```
    ledstate = LOW;
}
digitalWrite(led, ledstate);           // se activa el LED
suma = 0;                               // se inicializa la variable suma
}
```

```
// -----  
// SISTEMA DE TEMPERATURA  
// -----  
  
#include <OneWire.h>  
#include <DallasTemperature.h>  
#define Pin 5 // pin bus DATA  
OneWire ourWire(Pin); // pin comunicación OneWire  
DallasTemperature sensors(&ourWire); // librería DallasTemperature  
int fan = 2;  
int estado = LOW; // estado ventilador  
float temp = 0; // valor temperatura  
  
void compruebatemperatura(){  
  sensors.requestTemperatures();  
  temp = sensors.getTempCByIndex(0);  
  Serial.print(temp);  
  Serial.println(" grados Centigrados");  
  delay(500);  
  if ( temp > 27 ){  
    estado = HIGH;  
  } else if ( temp < 25 ){  
    estado = LOW ;  
  }  
  digitalWrite(fan, estado);  
}
```

```

// -----
// SISTEMA DE ILUMINACIÓN
// -----

int luminosidad = 0;           // valor de luminosidad
int luz = A5;
int rele = 4;
int relestate = LOW;         // activación relé
int umbral = 650;           // umbral inicial
char cadena[30];             // caché 30 registros
byte posicion=0;            // posición caché

void compruebaluz(){
  luminosidad = analogRead (luz);   // lectura luminosidad
  Serial.print(luminosidad);        // impresión valor
  Serial.println("  luminosidad");
  // encendido antorcha LED
  if (luminosidad < umbral ){
    relestate = HIGH;
  } else {
    relestate = LOW;
  }
  digitalWrite(rele, relestate);
  delay(100);
  // lectura datos buffer
  if(Serial.available())
  {
    memset(cadena,0,sizeof(cadena)); // borra array desde 0 hasta final
    while(Serial.available()>0)     // mientras hay datos ejecuta
    {
      delay(5);                     // pausa mejora recepción datos
      cadena[posicion]=Serial.read(); // lee carácter
      posicion++;
    }
    // conversión a caracteres enteros
    umbral = atoi(cadena);
    Serial.print(umbral);
    delay(300);
  }
  Serial.println(umbral);
}

```

```
    posicion = 0;  
}
```

```
// -----  
// PUESTA EN MARCHA DEL ARDUINO  
// -----  
  
void setup() {  
  // sistema temperatura  
  delay(500); // esperar medio segundo  
  Serial.begin(9600); // comunicación puerto serie  
  sensors.begin(); // sensores  
  pinMode(fan, OUTPUT); // pin ventilador  
  // sistema luz  
  pinMode(rele, OUTPUT); // pin relé  
  pinMode(luz, INPUT); // pin antorcha led  
  // sistema detector final filamento  
  pinMode(led, OUTPUT); // pin relé  
}  
  
void loop() { //se ejecuta el loop y se llama a cada una de las funciones  
  detectorfilamento();  
  compruebaLuz();  
  compruebatemperatura();  
}
```