

# IMPLEMENTACIÓN DE PROTOTIPO DE UNA MÁQUINA DE GRABADO CON LÁSER CONTROLADA POR UN RASPBERRY PI

***Horacio Orozco Mendoza***

Instituto Tecnológico de Celaya

*horacio.orozco@itcelaya.edu.mx*

***José Guadalupe Zavala Villalpando***

Instituto Tecnológico de Celaya

*jk.zavala@itcelaya.edu.mx*

***Karla Alejandra Rodríguez Sotelo***

Instituto Tecnológico de Celaya

*11030563@itcelaya.edu.mx*

***Omar Calderón Nieto***

Instituto Tecnológico de Celaya

*11251263@itcelaya.edu.mx*

## **Resumen**

En este trabajo se presenta una máquina PEN PLOTTER (EDG2400) para su uso como CNC de grabado con láser con una potencia de 2 W, cuya precisión es superior a 0.01 mm y con dimensiones de área de trabajo de 40x25 cm. La rehabilitación de la máquina PEN PLOTTER se realizó dentro de un esquema de bajo costo sin contemplar mano de obra.

Para el control de la máquina se hizo uso de una tarjeta Raspberry Pi con la finalidad de hacer portable el sistema sin sacrificar su funcionalidad. Con esto se logró que la máquina CNC no dependa de una computadora adicional, ya que la mayoría de las máquinas actuales utilizan diferentes tipos de placas de programación y éstas se conectan por medio de periféricos a la computadora impidiendo así su portabilidad.

Una de las múltiples ventajas que se tiene al utilizar la tarjeta Raspberry Pi son sus puertos GPIO, siendo estos de dos tipos, de propósitos generales (conexiones configurables) o especiales (conexiones seriales); teniendo de esta manera diversidad en la elección de comunicación, teniendo con esto una interfaz entre el programa y la parte electrónica.

Se hace uso del sistema operativo Ubuntu Mate en la Raspberry Pi para el desarrollo de una aplicación en lenguaje Python, que permite realizar el control de los motores a pasos con una etapa potencia adecuada por medio de los puertos periféricos (GPIO) generando la trayectoria y velocidad que el sistema requiera. El programa desarrollado tiene como finalidad el ejecutar los comandos proporcionados por un archivo Gcode, el cual contiene la vectorización de los patrones a ejecutar.

**Palabra(s) Clave(s):** CNC, Gcode, láser, pen plotter, raspberry Pi.

## **1. Introducción**

Las máquinas CNC con láser en su mayoría provienen de otros países, usualmente requieren para su funcionamiento del uso de una computadora y en algunos casos es necesario adquirir un software para su funcionamiento haciendo su adquisición más costosa.

Por otra parte para el desarrollo del sistema se propone la rehabilitación de una máquina PEN PLOTTER para uso como CNC de grabado con láser controlada por medio de una tarjeta Raspberry Pi. Para el uso de la máquina se desarrolló un programa para la plataforma Raspberry Pi con la finalidad de eliminar el uso de PC y tarjetas alternativas para el control de estas máquinas. Dicho programa está dirigido a la creación de una conexión innovadora de acuerdo a la tendencia de los lenguajes de programación.

## **2. Desarrollo**

El proyecto se encuentra dividido en tres secciones: procesamiento que engloba los softwares utilizados; la sección de control, donde se presentan las características de los motores y Driver utilizados; finalizando con la sección de

estructura, en donde se exponen los diseños creados para la implementación de la máquina PEN PLOTTER como grabado con láser.

## Procesamiento

En la programación para la implementación de este proyecto se usaron diferentes softwares, cada uno con una tarea específica como la generación del Gcode, la realización de una interfaz gráfica y la lectura del Gcode, todo ello bajo el sistema operativo Ubuntu Mate.

## Sistema operativo

Ubuntu Mate es un sistema operativo basado en Linux, esta versión está hecha para un procesador ARMv7, el cual tiene integrado la Raspberry Pi en sus modelos 2 y 3. Este sistema operativo se puede instalar en una memoria MicroSD de 8 Gb o mayor para ser utilizado en dicho ordenador. Se recomienda que sea en una memoria de mayor capacidad ya que tendrá un mejor rendimiento.

Las características de la tarjeta Raspberry Pi 2 utilizada en este proyecto se muestran en la tabla 1.

Tabla 1 Características Raspberry Pi 2.

Características	Descripción
Procesador	900MHz quad-core ARM Cortex-A7 CPU
RAM	1 GB
USB	4 Puertos
GPIO	40 Pines
HDMI	1 puerto
Ethernet	1 puerto
Audio	Jack de audio 3.5mm vídeo compuesto
Cámara	Interfaz de cámara (CSI)
MicroSD	1 lector
Video	VideoCore IV 3D

## Software

Los softwares utilizados son de acceso libre y corresponden a los siguientes:

## Python

Se hizo uso de Python que es un lenguaje de programación adoc para ser implementado en la tarjeta Raspberry Pi. Cuenta con estructuras de datos eficientes y de alto nivel y un enfoque simple pero efectivo a la programación orientada a objetos. Algunas de las características de este lenguaje se muestran en la tabla 2.

Tabla 2 Características lenguaje Python.

Características	Descripción
Propósito general	Se pueden crear todo tipo de programas. No es un lenguaje creado específicamente para la web.
Multiplataforma	Hay versiones disponibles en muchos sistemas informáticos distintos.
Interactivo	Dispone de un intérprete por línea de comandos en el que se pueden introducir sentencias.
Orientado a Objetos	Ofrece en muchos casos una manera sencilla de crear programas con componentes reutilizables.
Funciones y librerías	Dispone de muchas funciones incorporadas en el propio lenguaje, para el tratamiento de strings, números, archivos, etc.
Sintaxis clara	Posee una sintaxis muy visual, gracias a una notación indentada (con márgenes) de obligado cumplimiento.

## Qt Creator

Para el desarrollo de la interfaz gráfica se hizo uso del programa Qt Creator (figura 1); el cual es una interfaz gráfica de usuario cuyo objetivo es proporcionar una plataforma cruzada, con un completo entorno de desarrollo integrado (IDE) para desarrollar proyectos de Qt. Está disponible para Linux, Mac OS X y las plataformas de Windows.



Figura 1. Logo Qt Creator.

## **Inkscape**

Para la vectorización y creación del Gcode se hizo uso del programa Inkscape (figura 2), que es un editor gráfico de vectores, gratuito y de código libre, puede crear y editar gráficos vectoriales como ilustraciones, diagramas, líneas, gráficos, logotipos, e ilustraciones complejas. Dicho programa contiene sofisticadas herramientas de dibujo con una capacidad comparable a Adobe Illustrator, CorelDRAW y Xara Xtreme.



Figura 2 Logo Inkscape.

Inkscape se encuentra desarrollado principalmente para el sistema operativo GNU/Linux, pero es una herramienta multiplataforma y funciona en Windows, Mac OS X, y otros sistemas derivados de Unix.

### **Diagrama a bloques del programa**

En las figuras 3, 4 y 5 se muestra el diagrama a bloques de la lógica que se siguió para la realización del programa en lenguaje Python.

En la figura 3 se muestra la primera sección donde se ejecuta el programa, se calibra el láser dependiendo del material con el que se vaya a trabajar (MDF, vinil, acrílico, etc), se carga el archivo Gcode con extensión (.txt) si el archivo no existe se envía una advertencia. Se debe ingresar la velocidad de trabajo (el programa está limitado a 50mm/s) y la potencia del láser recordando que es de 2W (el programa está limitado a 90%); se puede empezar a leer el archivo Gcode desde cero o desde una línea que se introduzca, si se desea desde una línea específica esta deberá ser menor al número total de líneas del archivo. Una vez ingresada o no ingresada la línea de comienzo, el láser se posiciona en el origen y se comienza a leer línea a línea el archivo.

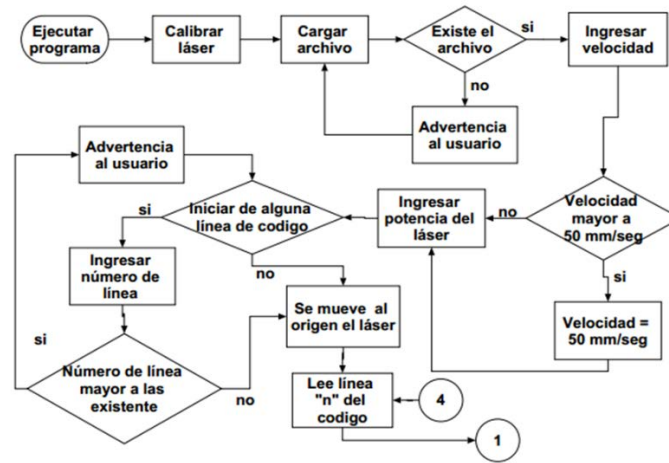


Figura 3 Diagrama a bloques.

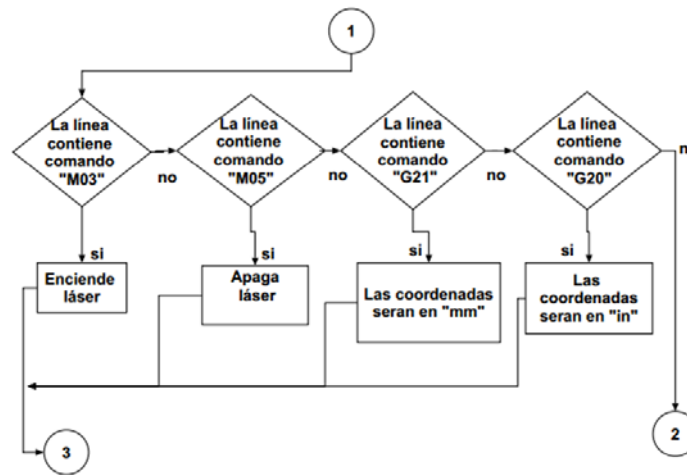


Figura 4 Diagrama a bloques.

Dentro del archivo Gcode se buscan los comando que nos interesan y se realizan las acciones correspondientes para cada uno, M03 (encender láser), M05 (apagar láser), G21 (coordenadas en mm), G20 (coordenadas en in) (figura 4).

Si los comandos leídos son G02 o G03 se extraen las coordenadas de X, Y, I y J respectivamente ya que este movimiento será circular; de lo contrario si se leen comandos G00 o G01 se extraen solamente X y Y debido a que el movimiento es lineal.

Si se encontró algún comando o se realizó el movimiento correspondiente se suma un 1 a la línea y se compara el número de líneas total con el leído, si son iguales el programa se termina si no es así se vuelve a iterar (figura 5).

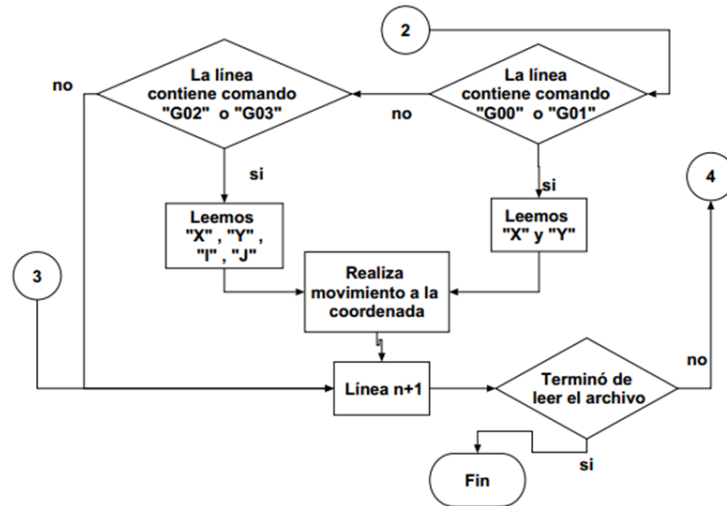


Figura 5 Programa a bloques.

## GPIO'S

La Raspberry Pi 2 cuenta con 40 pines como se muestra en la figura 6 de los cuales 2 son de voltaje a 3.3 V, 2 pines más a 5 V, 8 pines conectados a GND, 17 pines de uso general, 5 con comunicación de protocolo SPI, 2 para protocolo i2c, y 2 más de protocolo UART con 2 pines que no se utilizan. Para este proyecto se hizo uso de las terminales del GPIO que se muestran en tablas 3 y 4.

Tabla 3 GPIO's.

3.3V	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
DNC	27	28	DNC
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

Tabla 4 GPIO'S utilizados.

GPIO	# Pin	Actividad
17	11	Control láser
5	29	Sensor Y
6	31	Sensor X
18	12	Habilitar dirección Y
23	16	Pasos Y
24	18	Dirección Y
12	32	Botón Paro
16	36	Habilitar dirección X
20	38	Pasos X
21	40	Dirección X
26	37	Switcheo Voltaje

## Control

Para realizar el movimiento del trazado de las imágenes vectorizadas leídas por medio del programa en Python, se utilizaron motores cuyas características principales son: precisión y control de posición sin necesidad de sensores adicionales.

### Motores a paso bipolares

Los motores paso a paso bipolares tienen generalmente 4 cables de salida (figura 6). Necesitan ciertos trucos (voltajes bajos, corrientes altas y frecuencia de conmutación elevada) para ser controlados debido a que requieren del cambio de



dirección de flujo de corriente a través de las bobinas en la secuencia apropiada para realizar un movimiento.

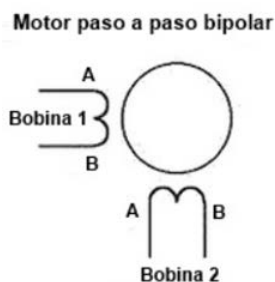


Figura 6 Motores paso a paso bipolares.

En la tabla 5 se muestra la secuencia necesaria para controlar motores paso a paso bipolares.

Tabla 5 Secuencia de control.

Paso	Terminales			
	A1	B1	A2	B2
1	+V	-V	+V	-V
2	+V	-V	-V	+V
3	-V	+V	-V	+V
4	-V	+V	+V	-V

En la selección de un motor paso a paso se tienen que tomar en cuenta ciertos factores dependiendo de la aplicación:

- Voltaje y corriente máximos
- Resolución
- Torque
- Costo

Las características de los motores que se utilizaron en este proyecto se muestran en tabla 6.

Tabla 6 Características motores.

Motor 1	Características	Motor 2
KP42EM1-016	Modelo	KP42HM1-022
Si	Bipolar (4 hilos)	Si

0.9	<b>Grados</b>	0.9
3-5 V	<b>Alimentación</b>	3-5V
0.5-1 A	<b>Corriente</b>	1-2 A

## Driver

Un driver es un conjunto de dispositivos electrónicos que recibe las órdenes de movimiento del ordenador y maneja en consecuencia los motores como se muestra en la figura 7.

Existe una amplia variedad de drivers que se pueden adquirir de acuerdo a sus características técnicas para adaptarse mejor a diferentes equipos.

Para el proyecto se desarrolló un driver con diversos componentes electrónicos con la intención de reducir el costo de la máquina.

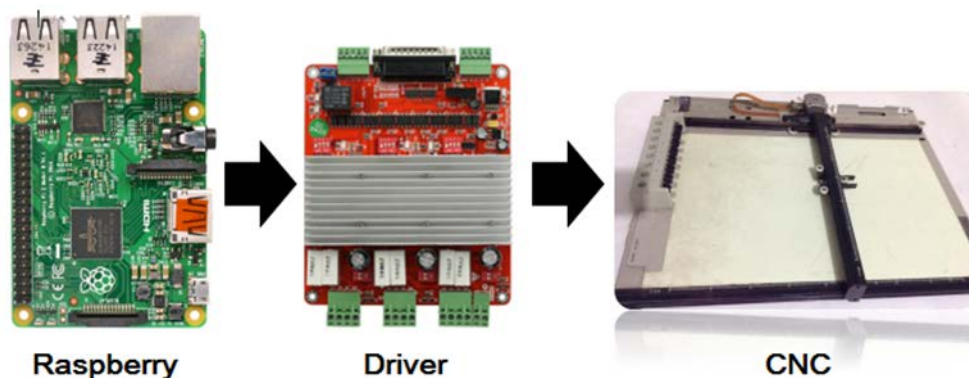


Figura 7 Esquema de comunicación.

El controlador utilizado está basado en el chip A4988 de Allegro (figura 9), cuyas características son:

- Control simple de dirección y paso.
- Cinco diferentes resoluciones de paso.
- Corriente de salida ajustable mediante un potenciómetro.
- Puede proporcionar hasta 2A por bobinado y opera desde 8 a 35V.
- Protección de temperatura, sobre-voltaje y sobre-corriente.
- Tienes salidas auto-reguladas de 5V y 3.3V para alimentar la lógica externa.

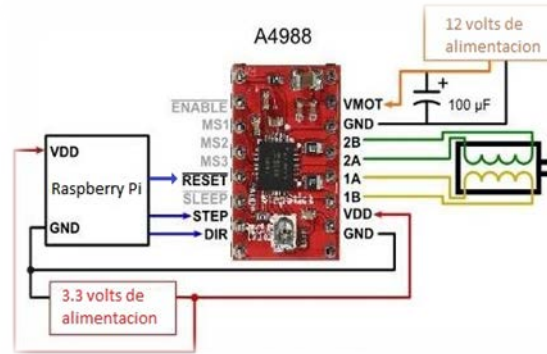


Figura 8 Driver A4988.

## Estructura

Como ya se mencionó se utilizó la estructura una máquina PEN PLOTTER. Un PEN PLOTTER realiza trazos por medio del movimiento de un bolígrafo u otro instrumento a través de la superficie de una hoja de papel. Esto significa que son dispositivos trazadores gráficos vectoriales. Pueden dibujar líneas de arte complejo, incluyendo el texto, pero lo hacen lentamente debido al movimiento mecánico de las plumas.

Para la finalidad de este proyecto se realizaron 3 prototipos de bases, debido a que se reemplazó el uso de un bolígrafo por un láser. En las figuras 9, 10 y 11 se muestran dichos prototipos, los cuales se diseñaron en el programa CAD (Dibujo Asistido por Computadora) para la base del láser que se adaptó a la estructura del PEN PLOTTER.

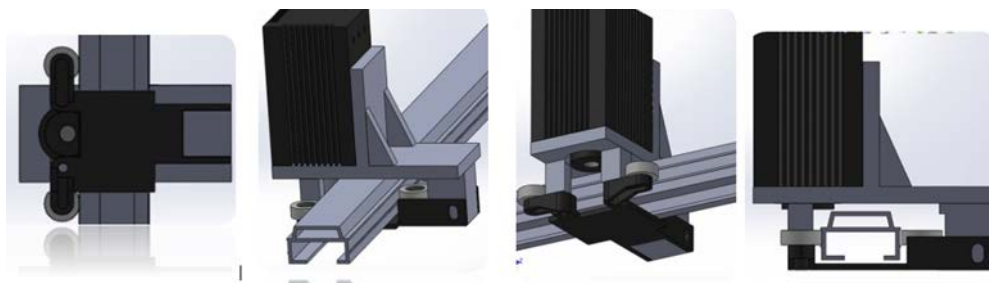


Figura 9 Prototipo 1.

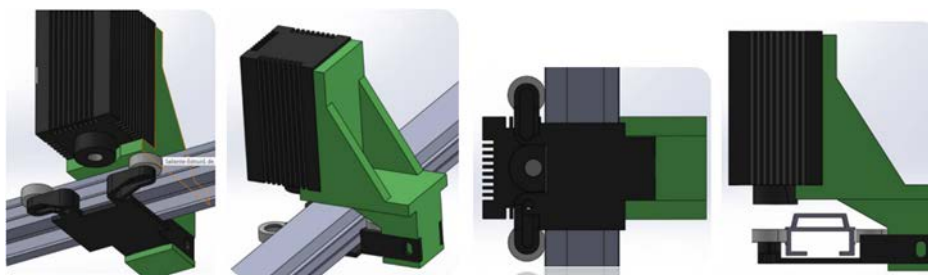


Figura 10 Prototipo 2.

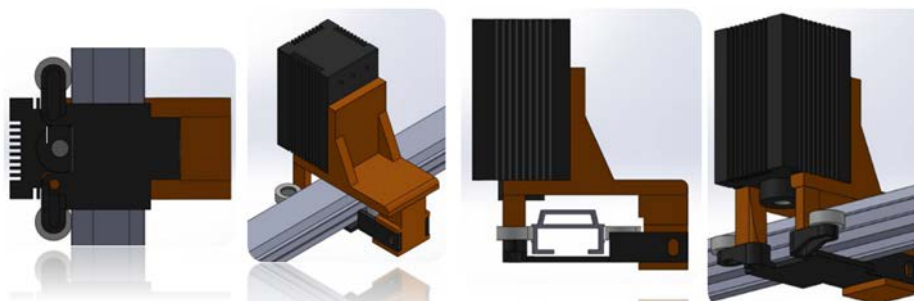


Figura 11 Prototipo 3.

El prototipo elegido para la realización de este proyecto fue el diseño 3, dicha selección se realizó en base a los criterios mostrados en la tabla 7.

Tabla 7 Tabla comparativa de prototipos.

Características	Prototipo 1	Prototipo 2	Prototipo 3
Acceso a la lente	Baja	Buena	Buena
Distribución de peso	Bueno	Bajo	Bueno

### 3. Resultados

Los resultados obtenidos a lo largo de este trabajo se mencionan a continuación. Se realizaron dos PCBs para el correcto funcionamiento del trabajo, una para la etapa del control de los motores (figura 12) y otra PCB (figura 13) para la alimentación de los motores y del láser.



Figura 12 PCB motores.



Figura 13 PCB alimentación.

Se realizó un programa de código abierto en el lenguaje de programación Python (figura 14).

Se realizó una interfaz gráfica (figura 15) para el manejo y control de la máquina CNC, dicha interfaz se diseñó en el programa Qt Creator utilizando su aplicación Qt Designer.

```

from PyQt4.QtCore import QTimer
import RPi.GPIO as GPIO
import math
GPIO.setmode(GPIO.BCM)
btnstop=12
btnx=6
btny=5
GPIO.setup(btnx,GPIO.IN)
GPIO.setup(btny,GPIO.IN)
GPIO.setup(btnstop,GPIO.IN)
pines=[21,20,16,24,23,18,17,26]
GPIO.setup(pines,GPIO.OUT)
GPIO.output(pines,False)
pwm = GPIO.PWM(17,500)
pwm.start(0)
pwm.ChangeDutyCycle(0)
GPIO.output(26,1)

#
form_class = uic.loadUiType("visualcncclaser.ui")[0]
#
class ThC(threading.Thread):
    def __init__(self,MyQueue,nombre,velocidad,opcional,potencia):
        threading.Thread.__init__(self)
        self.coorxQueue=MyQueue
        self.cooryQueue=MyQueue
        self.prxQueue=MyQueue
        self.pryQueue=MyQueue
        self.lineaQueue=MyQueue
        self.porcQueue=MyQueue
        self.laserQueue=MyQueue
        self.finQueue=MyQueue
        self.nombre=nombre
        self.velocidad=float(velocidad)
        self.opcional=opcional
        self.potencia=float(potencia)
        self.capturing = 0
        self.laser=0

```

Figura 14 Código Python.



Figura 15 Interfaz gráfica.

Algunos patrones obtenidos al poner en funcionamiento la máquina CNC láser se muestran en las figuras 16 y 17, en donde se puede observar la gran diversidad de grabados realizados.



Figura 16 Patrón realizado.



Figura 17 Patrón realizado.

#### 4. Conclusiones

Las conclusiones obtenidas en el desarrollo de este trabajo se enuncian a continuación, de tal forma que el proyecto se realizó de manera exitosa.

Se logró crear un código de código abierto en el lenguaje de programación Python. La máquina CNC de grabado en láser realiza patrones simples y complejos de forma óptima cuya precisión es superior a 0.01 mm.

El presupuesto inicial que se tenía planeado para la rehabilitación del PEN PLOTTER no fue respetado debido a que inicialmente se había contemplado un láser de menor potencia.

Se logró obtener total portabilidad del sistema por medio del control e implementación de la tarjeta Raspberry Pi.

Se diseñó e implementó una interfaz gráfica para el manejo y control de la máquina CNC de tal forma que fuese de fácil acceso para el usuario.

## **5. Referencias**

- [1] Raspberry Pi Model B. Fecha de consulta: 20 de mayo de 2016. URL: <https://www.raspberrypi.org/>.
- [2] About Python. Fecha de consulta: 20 de mayo de 2016. URL: <https://www.python.org/>.
- [3] Why Qt? Fecha de consulta: 20 de mayo de 2016. URL: <https://www.qt.io/ide/>.