

“Diseño de rutinas de control empleando sistema Arduino para el uso de diferentes extrusoras en impresión 3D, aplicación práctica en prototipo de impresión 3D”

Jaime Bustamante⁽¹⁾, Iván Salazar⁽²⁾, Carlos Valdivieso⁽³⁾
Facultad de Ingeniería en Electricidad y Computación
Escuela Superior Politécnica del Litoral (ESPOL)
Campus Gustavo Galindo, Km 30.5 vía Perimetral
Apartado 09-01-5863. Guayaquil-Ecuador
obustama@espol.edu.ec⁽¹⁾, isalazar@espol.edu.ec⁽²⁾, cvaldiv@fiec.espol.edu.ec⁽³⁾

Resumen

En el siguiente artículo se presenta el diseño de un prototipo el cual simulará un cabezal con 3 extrusoras las cuales deberán posicionarse según el color deseado gracias a algunas rutinas de control programadas, el control se hará mediante una tarjeta electrónica conocida como Arduino cuyo cerebro es un microcontrolador. Esta tarjeta es conocida como hardware libre al igual que su software, lo que permite que se realicen modificaciones de diseño en la tarjeta y su software es gratuito. La tarjeta Arduino también controlará el funcionamiento de una impresora 3D, con la ayuda de otra tarjeta electrónica llamada RAMPS. Esta tarjeta será la encargada de controlar el funcionamiento de cada uno de los componentes que integran la impresora 3D y que además llevará consigo unos drivers para motores de paso conocidos como POLOLU. Se detallarán todos los pasos a seguir en el proceso de impresión de un objeto en 3D con la ayuda de diagramas de bloques y se explicará el uso del programa principal con el cual visualizaremos parámetros, detalles de la impresión y el proceso de elaboración capa por capa.

Palabras Claves: Arduino, RAMPS, POLOLU, Microcontrolador, rutinas de control, impresora 3D.

Abstract

The following article shows a prototype design which simulates a rotating head with 3 extruders. These extruders will reach the desired position based on a known color through some programmed control routines, we will exercise control by an electronic board known as Arduino, using a micro controller as a brain. This is a free hardware board, as well as its software, which means we can make any changes on board's original design and load the software for free every time. Arduino will also control the 3D Printer functions with another electronic board, called RAMPS as a support. This second board will be responsible for controlling all the components that integrate the 3D printer. Besides, this board will carry some stepper motor drivers known as POLOLU. We will explain all the steps in the process to print a 3D object with block-based diagrams and we will explain how to use the main program that will show us parameters, printing details and the printing process layer by layer.

Keywords: Arduino, RAMPS, POLOLU, Microcontroller, control routines, 3D printer.

1. Introducción

El siguiente proyecto tiene como finalidad mostrar los principios básicos del ensamblaje, calibración y funcionamiento de la impresora Prusa Mendel modelo 2013 que imprime objetos en 3 dimensiones. Dentro del proceso de impresión se va a presentar la tarjeta electrónica Arduino, la cual contiene un microcontrolador que realizará todas las funciones necesarias para llevar a cabo la construcción de nuestro objeto en 3D. [5]

Se detallarán todos los programas necesarios para la ejecución de la impresión tanto para la tarjeta Arduino como para la interfaz con el usuario.

Aplicar conocimientos de programación en C++ es necesario para la programación y configuración de parámetros si se realizan ciertos cambios de algunos elementos del modelo original de la impresora. Basándonos en las limitaciones de nuestro modelo se piensa en una idea muy útil, es por esto que se realiza un proyecto el cual consiste en modificar el funcionamiento que tiene la extrusora principal y que aumenta las opciones de colores para imprimir un objeto. Un cabezal tendrá 3 extrusoras lo cual va a permitir seleccionar un color con el cual deseo crear mi objeto. Técnicas para controlar el giro milimétrico de un motor de paso bipolar serán aplicadas.

2. Fundamento teórico

La parte central de una impresora 3D tiene la idea de un robot cartesiano. Esta es una máquina que se puede mover en 3 direcciones lineales, a través de los ejes X, Y y Z, también conocidas como coordenadas cartesianas. Para hacer esto la impresora utiliza unos pequeños motores de paso que se pueden mover con bastante precisión y exactitud, usualmente 1.8° por paso, lo cual se traduce a un rango de resolución de fracciones de milímetros y que es la única forma a través de la cual estos motores de paso pueden ser controlados. [1]

La impresora que utilizaremos en la Prusa Mendel la cuál es de las más económicas y didácticas del mercado ya que nos permite armar la impresora, cambiar sus partes dañadas y ver cómo actúan sus piezas cuando está en funcionamiento. Posee 5 motores, uno para el movimiento del eje X, uno para el eje Y, dos para el eje Z y uno para la extrusora. Además tiene 3 sensores de final de carrera para cada eje y que también sirven para fijar un punto de referencia (home) en nuestro sistema.

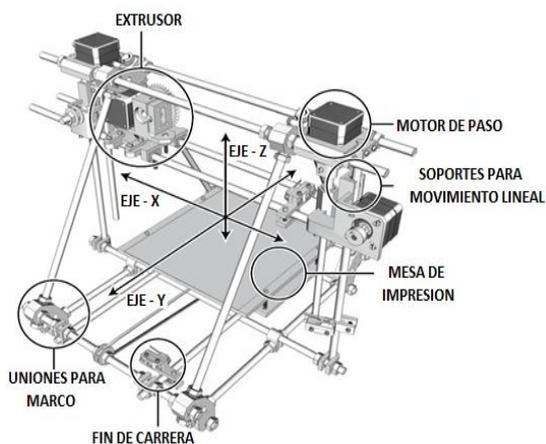


Figura 2.1 Impresora 3D Prusa Mendel DIY

El Arduino Mega es una tarjeta microcontroladora basada en el ATmega2560, tiene 54 pines de entradas/salidas digitales (de las cuales 15 pueden ser usadas como salidas PWM), 16 entradas analógicas, 4 UARTS (puertos serial para hardware), un cristal oscilador de 16MHz, conexión USB, entrada de corriente, conector ICSP y botón de reset. [2]

El Arduino Mega se puede programar con el software Arduino. El ATmega2560 en el Arduino Mega viene precargado con un gestor de arranque (bootloader) que permite cargar nuevo código sin necesidad de un programador por hardware externo. Se comunica utilizando el protocolo STK500 original.

También podemos omitir el gestor de arranque y programar directamente el microcontrolador a través del puerto ICSP (In Circuit Serial Programming). [3]



Figura 2.2 Tarjeta Arduino Mega

La RepRap Arduino Mega Pololu Shield conocida también como RAMPS es una tarjeta controladora que trabaja en conjunto con la tarjeta microcontroladora Arduino Mega.

RAMPS es una solución de dos tarjetas que consiste en una tarjeta microcontroladora de Arduino Mega y una placa especialmente diseñada que se conecta encima del Arduino. El Arduino mega proporciona el cerebro a la plataforma, mientras que la placa proporciona el hardware de conmutación.

- Soporta hasta 5 tarjetas controladoras para motores de paso Pololu 4988 o similares.
- 3 conmutadores de carga de alta potencia fusionados a 5amp y 11 amp para la extrusora.
- 6 conexiones para final de carrera.
- 3 conexiones para termistores.
- Una entrada de alimentación dual de 12 a 3 voltios a 16 amp como máximo.



Figura 2.3 Tarjeta Arduino Mega y la RAMPS

La electrónica de nuestra impresora 3D incluye muchas partes diferentes, trabajando juntas, para construir las impresiones 3D. Estos componentes incluyen un microcontrolador, una tarjeta principal, controladores de motores y motores de paso, una boquilla de extrusión y partes como finales de carrera o interruptores de límite y sensores de temperatura. Para tener una idea de cómo estas piezas diferentes trabajan juntas observemos la figura 2.4 [1]

El microcontrolador, se encuentra en una tarjeta completamente separada, es una pequeña y simple computadora que ejecuta código especializado llamado firmware que le permite leer e interpretar sensores

como los de temperatura y final de carrera, así como el manejo de motores utilizando drivers de motores para conmutar altas cargas usando transistores para altas corrientes llamados MOSFET's. [1]

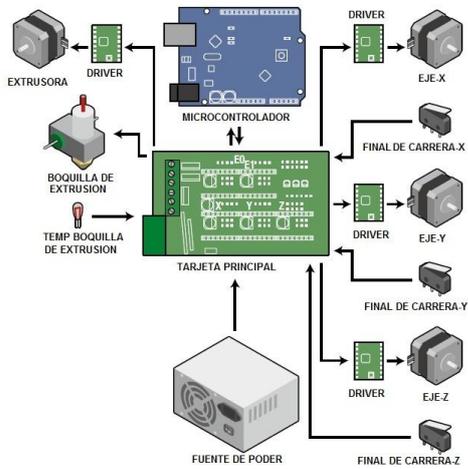


Figura 2.4 Electrónica de una impresora 3D.

Pronterface es el programa que va a permitir interactuar con la impresora, nos va a mostrar la configuración de los parámetros para imprimir nuestro objeto. Con la ayuda de este programa visualizaremos la construcción de nuestro objeto capa por capa y detalles como el porcentaje de construcción y tiempo estimado de elaboración entre otras cosas. Este programa acepta archivos .STL y .Gcode.

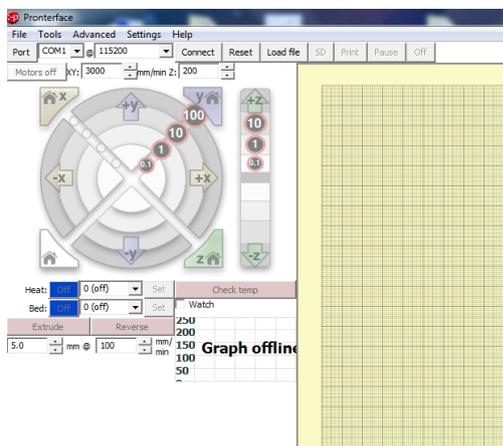


Figura 2.5 Pronterface software de comunicación

3. Ejercicios previos y realización del proyecto

En este capítulo detallamos ejercicios previos que nos ayudan a entender el funcionamiento de Arduino y aprender cómo funcionan ciertos comandos de la programación. Estos ejercicios aportan ciertos conocimientos para la realización del proyecto final.

3.1 Comunicación con la tarjeta Arduino.

En este ejercicio vamos a configurar 2 opciones básicas dentro del software de Arduino, para establecer una correcta conexión con la tarjeta y luego cargaremos un programa de prueba para verificar que el programa fue cargado correctamente en el Arduino. Dentro del programa Arduino en la pestaña "Herramientas" debemos configurar las opciones "Tarjeta" y "Puerto Serial"; en "Tarjeta" seleccionamos el modelo de la tarjeta Arduino que poseemos, en este caso seleccionamos Arduino Mega 2560; en "Puerto Serial" debemos seleccionar el puerto en el cual está conectado nuestro Arduino revisando en "Administrador de Dispositivos".

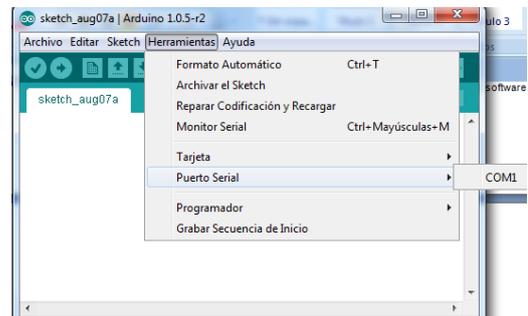


Figura 3.1 Pestaña Herramientas y Puerto Serial.

3.1.1. Ejemplo Blink.

Blink es un programa que va a poner a parpadear un Led de la tarjeta, con lo que vamos a comprobar si las configuraciones previas estuvieron correctas y se realiza la comunicación.

3.1.2. Diagrama de bloques

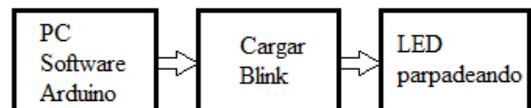


Figura 3.2 Diagrama de bloques del ejemplo "Blink"

3.2 Semáforo utilizando 3 colores de Led.

En el siguiente ejercicio mostraremos una de las aplicaciones que posee la tarjeta Arduino como lo es declarar puertos digitales como salidas, establecer los puertos digitales en una señal de ALTO o BAJO y también el uso de funciones temporizadoras que provocan retardos de milisegundos en la ejecución del programa.

Para esto vamos a utilizar 3 LEDs de 3 colores diferentes y 3 resistencias de 150ohm, la conexión la realizaremos tal como se muestra en la figura.

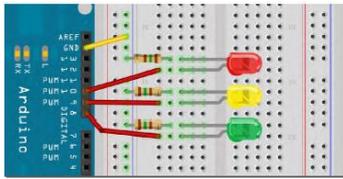


Figura 3.3 Esquemático del Semáforo.

3.2.1. Diagrama de bloques

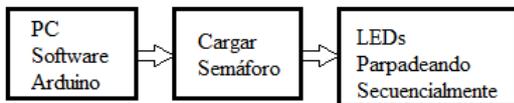


Figura 3.4 Diagrama de bloques del Semáforo.

3.3. Selector de 3 extrusoras mediante Arduino y L293D.

En este proyecto vamos a utilizar la impresora Reprap Prusa Mendel. Dentro de los cambios que vamos a realizar a esta impresora están:

- Reemplazar la tarjeta controladora original por una Arduino Mega 2560.
- Reemplazar los sensores ópticos de los finales de carrera por unos mecánicos.
- Reemplazar el firmware Sprinter por el firmware Marlin.
- Configuración de parámetros dentro de Marlin.

Por defecto el firmware de Marlin viene configurado para 115200 Baudrate nosotros deshabilitamos este valor y activamos 250000 Baudrate. Este cambio también lo realizamos en los parámetros de Pronterface antes de la primera impresión. Dentro de Marlin otro cambio que debemos realizar es la desactivación de la inversión de la lógica de funcionamiento de los finales de carrera. Todos estos cambios los visualizamos en la figura 3.5

```

// This determines the communication speed of t
#define BAUDRATE 250000
// #define BAUDRATE 115200

// The following define selects which electro
// 10 = Gen7 custom (Alfons3 Version) "https://
// 11 = Gen7 v1.1, v1.2 = 11

// The pullups are needed if you directly connect
const bool X_ENDSTOPS_INVERTING = false; // set
const bool Y_ENDSTOPS_INVERTING = false; // set
const bool Z_ENDSTOPS_INVERTING = false; // set
// #define DISABLE_MAX_ENDSTOPS

// For Inverting Stepper Enable Pins (Active Low

```

Figura 3.5 Configuración parámetros Marlin

Los parámetros de hardware que se deben calibrar antes de la primera impresión son:

- Nivelar la mesa de impresión en todo el plano para que no ocurran fallas en el proceso de impresión.
- Calibrar la distancia que debe tener la boquilla de la extrusora y la mesa de impresión (final de carrera eje Z).
- Verificar que al realizar la conexión a través de Pronterface el sensor de la extrusora marque en la pantalla la temperatura ambiente.
- Siempre se debe colocar cinta Kapton sobre la mesa de impresión para obtener una buena adherencia del material utilizado (PLA).
- Antes de imprimir verificar que nuestro diseño digital 3D este dentro de los parámetros de tamaño que determina nuestra impresora (20x20x20cm).

La selección de los colores (extrusoras) es controlada por un motor de paso bipolar mediante una tarjeta Arduino y dos drivers L293D. [4]

La preselección dependerá de las condiciones que se determinaron mediante el switch de 3 posiciones y las señales que son enviadas por los sensores ópticos cuando se mueve el cabezal, con todas esas condiciones tendremos el color que deseamos.

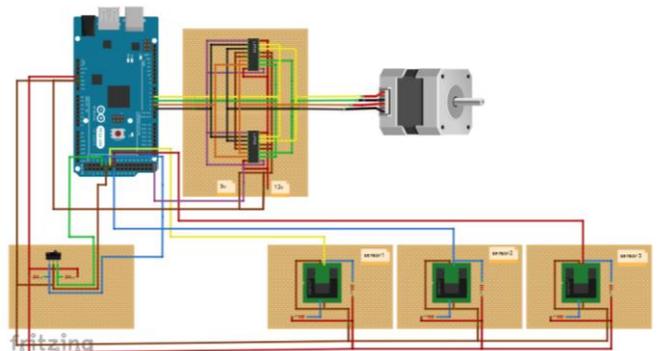


Figura 3.6 Esquemático del Selector de Extrusoras.

3.3.1. Diagrama de bloques

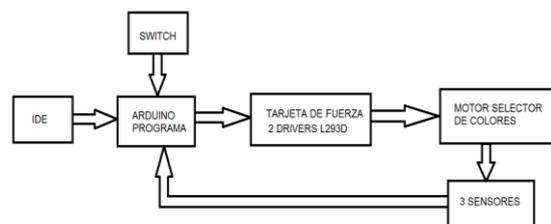


Figura 3.7 Diagrama de bloques Selector de 3 Extrusoras.

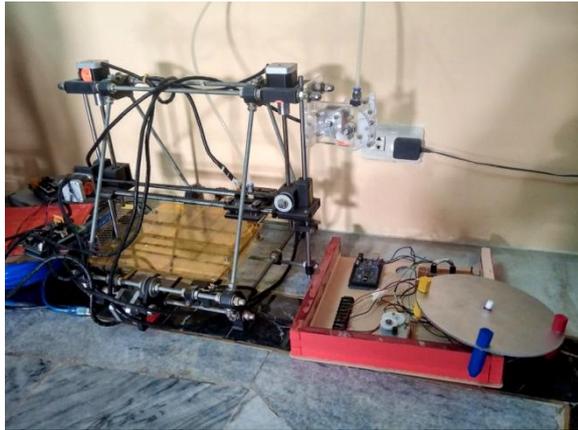


Figura 3.8 Impresora Prusa y Maqueta de 3 extrusoras

3.3.2 Descripción del algoritmo.

- Se inicializan las variables.
- Se configuran los puertos referentes a las variables que se están utilizando.
- Se ejecuta el programa:
 - Seteo el switch de 3 posiciones(X Y Z).
 - Verificación del estado de los sensores (1 2 3) dependiendo de la condición del switch.
 - Gira el motor para un determinado sentido por las condiciones previas.
 - Habilitará la salida del color elegido (amarillo, azul, rojo).

4. Implementación ejercicios de prueba y simulación del proyecto

En este último capítulo detallamos que componentes utilizamos para armar cada uno de los circuitos. Mostraremos imágenes de todos los ejercicios implementados previamente a la construcción de nuestro trabajo final. Observaciones para una correcta ejecución de cada uno de los ejercicios también serán explicadas en este capítulo.

Para el trabajo final adicionaremos un diagrama de conexiones de la maqueta final que se implementó para la simulación del selector de 3 extrusoras y los puertos de conexiones del microcontrolador ATmega 2560. Este diseño fue creado en el software Fritzing.

4.1. Comunicación con la Tarjeta Arduino.

Una vez que tenemos instalado nuestro IDE (software Arduino 1.0.5), seleccionamos el modelo de la tarjeta para que se instalen los drivers y finalmente escogemos el puerto COM al cual estamos conectados.

El programa Blink es uno de los programas básicos que vienen pre instalados con el software Arduino y que nos permite visualizar directamente, con un led en la tarjeta, que la comunicación y el programa fueron cargados exitosamente.

El LED de la tarjeta Arduino viene conectado internamente al Pin #13 es por esto que el código fuente debe utilizar sólo este pin.

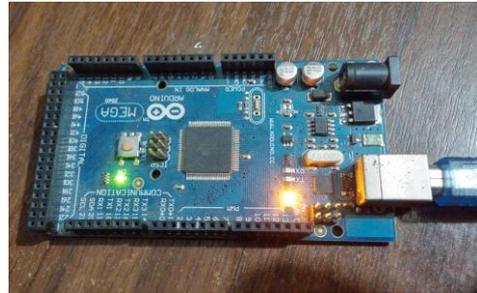


Figura 4.1 Comunicación con la placa Arduino.

4.1.1 Observaciones

El tiempo que dura el parpadeo del LED se puede modificar mediante el código fuente de Blink, este tiempo viene configurado inicialmente en 1segundo o 1000ms.

Durante la transmisión de cada programa se deben encender los LEDs de Rx y Tx y apagarse cuando finalice toda la carga.

4.2. Semáforo usando 3 colores de LED

En este proyecto vamos a simular el funcionamiento de un semáforo convencional en nuestro país pero va a trabajar de forma independiente sin la condición de un segundo semáforo.

Cuando se trabaja con diodos LED tenemos una polaridad (ánodo y cátodo) que es muy importante tener en cuenta para que estos se enciendan y no sufran daños, adicionalmente siempre que se trabaja con LED es necesario conectarles una resistencia para que fluya una corriente a través de ellos.

El semáforo que implementamos obedece directamente a la programación ya que en él vamos a determinar cuáles son los pines que vamos a utilizar como salidas digitales y también mediante la función "Delay()" o Retardo de tiempo, vamos a poder controlar los tiempos en que cada uno de los LEDs permanece encendido o apagado. El orden en que se encienden o se apagan los LEDs va a corresponder a la lógica de la programación y este también podrá ser modificado en caso de que queramos simular un semáforo de otro país.

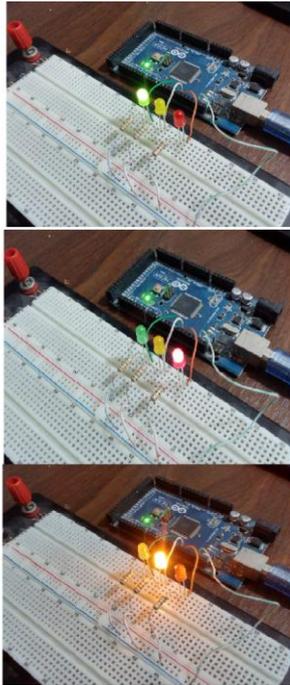


Figura 4.2 Semáforo tradicional de 3 LED temporizado.

4.2.1 Lista de Materiales

- 3 diodos LED color Rojo, Amarillo y Verde.
- 3 resistencias de 150 ohm.
- 1 protoboard.
- Alambres de cobre.

4.2.2 Observaciones

La función Delay() siempre toma valores en milisegundos por lo que se recomienda considerar valores mayores a 1000ms si queremos visualizar los cambios.

Se recomienda que el valor de la resistencia que se utilice no sea muy alto ya que esto baja la intensidad de luz de los LEDs.

No debemos olvidar que la tierra es un punto común y debe estar conectada al pin “GND” de nuestro Arduino, en caso de que uno de los LEDs no encienda se recomienda verificar la forma del punto común que tiene nuestro protoboard ya que esta puede ser una de las causas.

4.3. Selector de 3 extrusoras mediante Arduino y el driver L293D

El desarrollo de este proyecto tiene como finalidad realizar rutinas de control, basadas en una lógica del funcionamiento de 3 sensores ópticos, mediante las cuales podamos crear una nueva alternativa de selección entre 3 extrusoras, las cuales manejan 3

bobinas de PLA de diferente color; lo que nos permitiría escoger un color de nuestra preferencia para imprimir un objeto deseado en la impresora 3D. Adicionalmente a la utilización de la tarjeta Arduino ATmega 2560, hemos considerado necesario el uso de algunos elementos electrónicos adicionales los cuales nos ayudaron a que todo este proyecto se lleve a cabo. Para el funcionamiento del modelo base que es la impresora Reprap Prusa Mendel se deben realizar verificaciones y calibraciones previas que son necesarias antes de comenzar a imprimir, a continuación se detallan las principales:

- Correcto funcionamiento del termistor de la extrusora.
- Correcto drenaje de material a través de la boquilla de la extrusora, debe ser fluido y con una caída casi vertical.
- Correcta ubicación del final de carrera de todos los ejes principalmente del eje Z para mantener la mínima distancia entre la boquilla de la extrusora y la cama de impresión.
- Una correcta y precisa nivelación de la “cama” o base sobre la cual se van a asentar todos los modelos 3D.

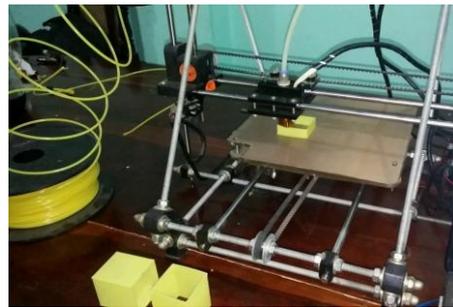


Figura 4.3 Impresión de un cubo de 4 caras.

Slic3r nos permite realizar modificaciones en la velocidad de impresión, es decir cuánto material va a extrudir por segundo, que tan caliente queremos que este la extrusora antes de empezar la impresión, entre otros. Todos estos parámetros van a ser claves para que nuestro objeto 3D tenga finos acabados y mayor detalle en sus medidas.

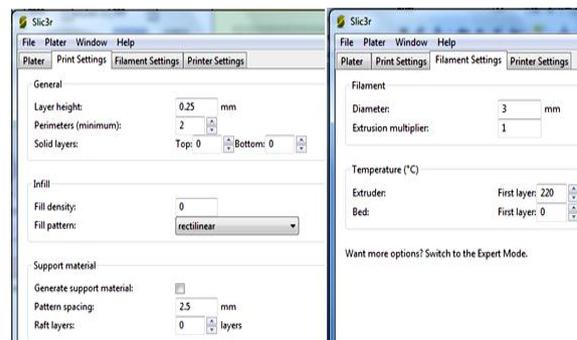


Figura 4.4 Parámetros de Slic3r

Debemos tener en cuenta que dependiendo del tamaño y la forma que tenga nuestro objeto, será más óptimo o menos óptimo el hecho de que realicemos cambios en la configuración inicial que trae Slic3r por defecto.

Como primeros ejercicios a imprimir es recomendable que se realicen objetos simples y de pequeño tamaño, objetos cuadrados como el cubo nos pueden servir como referencia para darnos cuenta de que nuestra impresora está alineada y que los finales de carrera están bien ubicados.

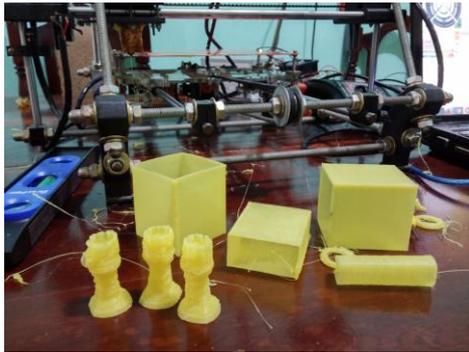


Figura 4.5 Primeros diseños 3D impresos.

Para el manejo de la potencia que necesitaba nuestro motor de paso bipolar fue necesaria la elaboración de una tarjeta controladora formada por 2 integrados L293D los cuales se reparten la potencia en partes iguales, evitando así el recalentamiento y posterior daño de estos integrados.

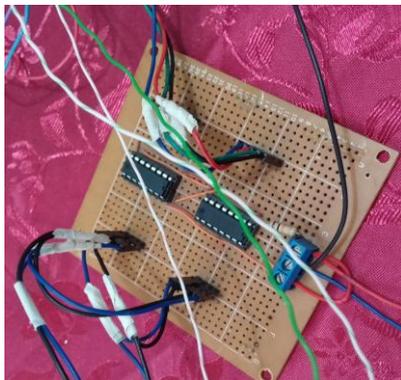


Figura 4.6 Tarjeta controladora de motor de paso formada con 2 integrados L293D

Una de las ideas claves para la elaboración de este selector de extrusoras fue la creación de un arreglo de 3 sensores. Estos sensores están ubicados de forma circular con una separación de 90° y en una cuarta posición tenemos un espacio en blanco que también es considerado. Se maneja una lógica de ON/OFF para cada uno de los sensores pero que requieren los estados de los demás para validar la posición final que está condicionada por la posición del Switch.

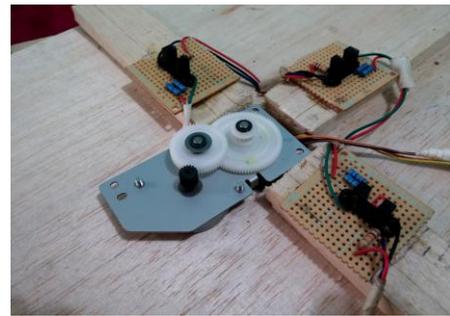


Figura 4.7 Sensores ópticos y motor de paso bipolar.



Figura 4.8 Maqueta del selector de extrusoras y la impresora Prusa Mendel 2013.

4.3.1 Listado de Materiales

- Tarjeta Arduino ATmega2560
- Cable Adaptador USB.
- 2 integrados L293D.
- 8 Resistencias de 1KΩ.
- 1 switch de tres estados.
- Cable UTP.
- Motor de paso bipolar de 12voltios
- Fuente de 12voltios.
- 3 Sensores ópticos.

4.3.2 Diagrama de conexiones.

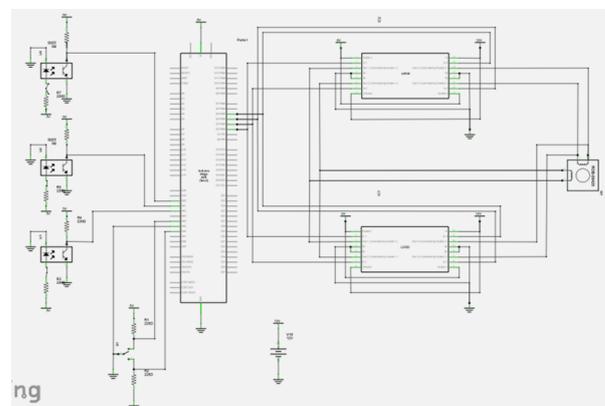


Figura 4.9 Selector de 3 extrusoras y el ATmega2560

4.3.3 Observaciones

Siempre verificar que la palanca que sirve para presionar el filamento este en la correcta posición para que haga presión en el material, ya que generalmente con el uso y movimiento de la maquina se afloja.

Si cuando mandamos a extrudir material no cae en forma vertical puede ser que existan trozos de material frio que estén adheridos en la punta de la boquilla por lo que debemos limpiarla y mandar a extrudir algunos cm de material hasta que nuevamente empiece a caer libremente en forma vertical.

La alineación de los sensores en la maquina debe ser con una separación de 90° entre los 3 ya que se pueden presentar problemas de posicionamiento de la extrusora si por la mala ubicación de estos el programa detecta un patrón erróneo y se detiene.

5. Conclusiones

- 1) Al ejecutar el programa en el Arduino, se observa que el disco donde están las extrusoras se comienza a mover, esto se debe a las condiciones de estado que maneja el switch. El usuario debe esperar hasta que la maquina termine su ejecución, es decir esperar hasta que el cabezal se detenga en el color que está seleccionado por el switch, en ese momento el operador podrá seleccionar que color quiere para su impresión.
- 2) La frecuencia que se maneja a nivel de software no nos permitía tener un giro lento del eje de motor para así obtener una mayor precisión en el posicionamiento de cada una de las extrusoras, es por esto que decidimos trabajar con un motor cuyo eje tenga adaptado un sistema de engranes con el cual pudimos reducir la velocidad angular del nuevo eje principal.
- 3) Dentro del control de la frecuencia de giro del cabezal se creyó conveniente programar la frecuencia, controlada con su inversa que es el tiempo, de tal manera que empiece con un valor y este vaya disminuyendo; es decir reduciendo su velocidad hasta un valor mínimo para que así cuando este próximo a los sensores tenga un giro lento y el frenado sea más preciso.
- 4) Cuando se trabaja con diseños complejos como el escorpión, es necesario aumentar la temperatura de la extrusora a 220°C y reducir la velocidad de extrusión del filamento ya que con esto logramos más precisión y mejor detalle en los acabados de nuestro modelo 3D a pesar de que aumenta considerablemente el tiempo de impresión.

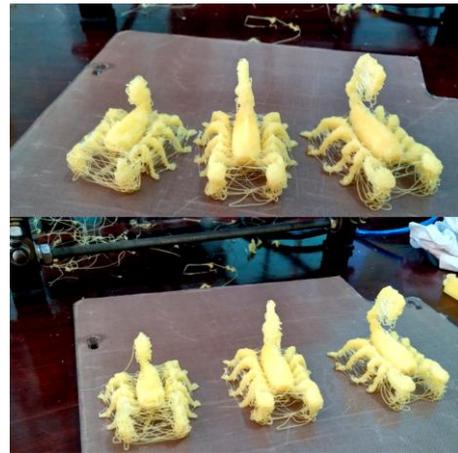


Figura 4.10 Escorpión impreso con diferentes configuraciones de velocidad y temperatura.

6. Recomendaciones

- 1) Es recomendable tener siempre conectado el ventilador para el Arduino y la RAMPS ya que los Pololu manejan corrientes considerables lo que hace que se recalienten y lleguen a valores muy elevados de temperatura.
- 2) Se recomienda siempre colocar cinta Kapton en la superficie sobre la cual va a descansar nuestro objeto mientras es construido, debido a que por la composición química del PLA obtenemos mejor adherencia y firmeza en la base, evitando que nuestro objeto 3D se despegue antes de finalizar la impresión. Debemos cambiarla cada 3 impresiones dependiendo del desgaste.

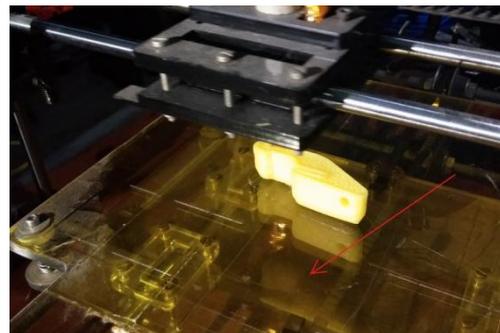


Figura 4.11 Cinta Kapton recubre la cama de impresión.

- 3) Se recomienda revisar constantemente la distancia entre la boquilla de extrusora y la cama sobre la que se asienta nuestro objeto, esta debe ser apenas 0,5mm para que existe una correcta adherencia del material y la cama. Esta distancia se la puede corregir deslizando ligeramente el interruptor de final de carrera del eje Z hacia arriba o hacia abajo según sea el caso.

7. Referencias

- [1] Brian Evans, Practical 3D Printers the Science and Art of 3D Printing, Springer Science Business Media, 2012.

- [2] Arduino SA, Arduino Mega 2560,
<http://arduino.cc/en/Main/arduinoBoardMega2560>,
fecha de consulta junio de 2013.

- [3] Simon Monk, Programming Arduino Getting Started with sketches, McGraw-Hill, 2012

- [4] Michael McRoberts, Beginning Arduino, Springer Science Business Media, 2010.

- [5] Arduino SA, ¿Qué es Arduino?,
<http://arduino.cc/es/Guide/Introduction>, fecha de consulta junio de 2013.