



aura

The word 'aura' is written in a large, bold, black font. The letters are semi-transparent, allowing a bright, glowing orange and yellow light to shine through from behind them. This light creates a lens flare effect and illuminates the top edge of the letters.

IES el Burgo de Las Rozas

Proyecto de bachillerato de excelencia

# Diseño, desarrollo y construcción del brazo robótico

## AURA

**Adrián de la Iglesia Valls**  
**2013**

A todo aquel que  
creyó que era posible

**Profesores colaboradores:**

Guzmán Naranjo

Cristina Jiménez

Eugenio García

## Introducción

La tecnología es una herramienta con un poder extraordinario, desde los bifaces a los robots, la tecnología cambia la manera que tenemos de concebir el mundo. La tecnología puede servir a muchas causas, no todas son buenas, pero existen personas decididas a entregar su vida a la tecnología para apoyar causas como la igualdad, con la tecnología de integración para discapacitados; la protección de las personas, con los sistemas de prevención de desastres naturales; y otros muchos ejemplos. Existen personas con estos ideales, dispuestos a usar todos sus recursos para cambiar el mundo, pero por falta de medios económicos no pudieron desarrollar su pasión por la tecnología. El dinero nunca debería ser una barrera para el aprendizaje.

Este proyecto está enfocado para que cualquier persona a la que le apasione la robótica, o la tecnología en general, tenga una plataforma para empezar a desarrollar su pasión sin limitaciones económicas. Poniendo en internet todo lo necesario para que se pueda construir un brazo robótico plenamente funcional en casa. AURA es un brazo robótico de bajo presupuesto, fácil de construir y programado para que su uso sea intuitivo.

Desde su diseño hasta su programación AURA pretende ser un brazo diferente, centrado en el usuario. Está diseñado en ordenador y construido por medio de una impresora 3D con una materia prima muy barata. El brazo está programado en un entorno de Arduino, totalmente gratuito. El programa de control utiliza recursos como el posicionamiento por coordenadas cilíndricas o el uso de un interfaz gráfico, para hacer su uso lo más intuitivo posible. Y todo, desde las piezas hasta los programas, pasando por toda la información necesaria para construir el brazo estará en internet sin ningún coste ni petición.

AURA es un brazo robótico para todos los usos y para todos los usuarios.



## Índice

|                                    |    |
|------------------------------------|----|
| Introducción .....                 | 1  |
| Objetivos .....                    | 3  |
| Descripción Técnica .....          | 4  |
| Descripción preliminar .....       | 4  |
| Método de diseño .....             | 4  |
| Descripción física de piezas ..... | 5  |
| Desarrollo matemático.....         | 13 |
| Programación.....                  | 19 |
| Interfaz y control .....           | 20 |
| Impresión 3D.....                  | 21 |
| Memoria .....                      | 23 |
| Presupuesto .....                  | 24 |
| Bibliografía.....                  | 25 |
| Anexos .....                       | 26 |

## Objetivos

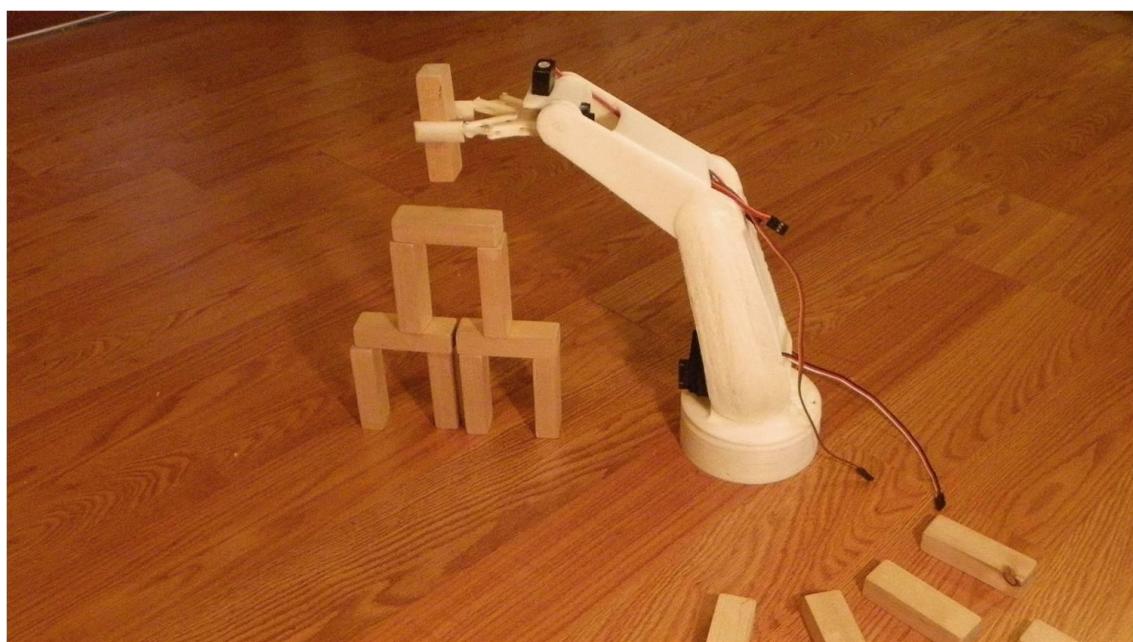
**Objetivo principal:** Diseñar y construir un brazo robótico simple de usar, casero y de bajo coste.

**Objetivo secundario:** Crear una plataforma de código abierto accesible a cualquier persona.

En internet existen una gran variedad de brazos robóticos y una infinidad de proyectos robóticos, sin embargo no existe ninguna plataforma accesible y de bajo precio para que cualquier persona que quiera iniciarse en el mundo de la robótica, de la programación o de la automatización pueda empezar sin grandes dificultades ni costes. Este proyecto comienza enfocado a diseñar y construir un brazo robótico plenamente funcional, fácil de usar y de bajo coste, pero pretende ir más allá. El brazo y el proyecto en su totalidad serán puestos en internet para el libre acceso de cualquier persona interesada sin ningún coste añadido.

Para conseguir el objetivo principal tenemos que ir consiguiendo una sucesión de objetivos intermedios en un orden adecuado. Primero tenemos que tomar una serie de decisiones como las dimensiones, las características técnicas que queremos tener o el método de construcción que queremos usar, en nuestro caso será la impresión 3D. Luego tendremos que diseñar el brazo al completo de forma que sea imprimible y funcional. Después habrá que construir el brazo y conseguir las piezas no imprimibles como los servos o la placa controladora. Finalmente tendremos que programar el brazo robótico y el interfaz de tal manera que sea intuitivo y fácil de manejar.

El segundo objetivo no tiene ninguna complicación en el proceso pero merece la pena explicar la importancia didáctica que posee. En un único sitio web podremos encontrar todas las piezas de un brazo robótico en distintos formatos: en un formato universal para que cualquier persona pueda descargárselo, verlo e imprimirlo y en el formato original para aquellos que tengan acceso al programa y quieran rediseñar el robot para adaptarlo a un uso específico, aportarle un acabado más estético y personal, o por cualquier otro motivo. En este sitio web también se va a poner el programa completo y el interfaz, además de información adicional como un manual de montaje, instrucciones para conseguir los programas necesarios de forma legal y gratuita, e incluso este mismo proyecto. Con todo, habremos creado una plataforma completa, abierta y gratuita para que todas las personas que quieran iniciarse en la robótica puedan tener un brazo para aprender y practicar diseño, programación o automatización.



## Descripción preliminar

AURA es un brazo robótico casero de tipo industrial con cinco grados de libertad. Tiene una mano con dos dedos en pinza con una abertura máxima de 6,84 cm. El espacio de acción máximo del brazo robótico es una semiesfera de radio 30 cm y el peso total de éste son 520g. El brazo está diseñado en su totalidad en un programa de modelado 3D, sus piezas han sido fabricadas mediante una impresora 3D que trabaja en ABS, un plástico barato y resistente. El robot debe sus movimientos y su posicionamiento a sus cinco servos, elegidos cada uno específicamente para la función que despeña. Estos servos están dirigidos desde la placa controladora del robot, un Arduino Leonardo que tiene cargado en su memoria un programa escrito en lenguaje C que interpreta los comandos entrantes y gestiona los servos para actuar en consecuencia. Los comandos destinados a controlar el brazo robótico salen del programa interfaz, un programa operativo desde cualquier ordenador que facilita la interacción del usuario con el robot mediante un sistema gráfico e intuitivo.



Para facilitar su control, la posición del brazo robótico en el espacio viene definida en un sistema de coordenadas cilíndricas. Dos coordenadas lineales, “d” es la distancia desde una posición hasta el centro de la base en un plano paralelo al suelo y “h” es la altura desde esta posición hasta el suelo; Y una coordenada angular “ $\omega$ ”, que es el ángulo entre la recta que une dicha posición con el centro de la base y una recta fija que tomamos como cero. También contaremos con una cuarta coordenada, “ $\theta$ ” es el ángulo que forma la mano con el suelo, que será por defecto cero para que la mano vaya paralela al suelo.

## Método de diseño

El brazo robótico AURA está ideado y diseñado para ser un brazo robótico diferente, con un aspecto distintivo al resto de brazos de su clase.

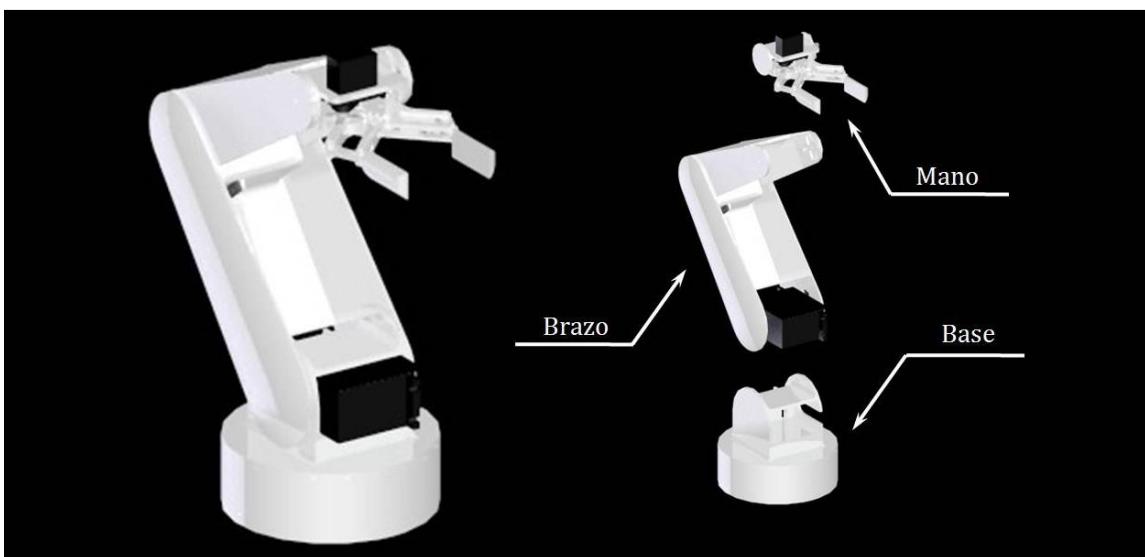
El brazo robótico en su totalidad está diseñado en Solid Edge. Éste es un programa de CAD o modelado 3D, especializado en el diseño mecánico para la creación de prototipos virtuales. Este programa aporta mejoras respecto a otros programas de CAD como la creación de conjuntos, que permite diseñar piezas individuales y juntarlas todas en un modelo; la sincronización de piezas, que permite realizar un cambio en una y que se actualice en todos los conjuntos en que aparezca; distintos métodos de diseño que simulan métodos de fabricación; y otras muchas ayudas al diseño que perfilan a Solid Edge como un programa CAD enfocado a ingenieros y a usos industriales. También se puede elegir el material del objeto a diseñar consiguiendo información del peso que tendrá, su resistencia, su aspecto y otros datos de utilidad.

Para crear el brazo fue necesario diseñar individualmente cada pieza por separado y unirlas después en un conjunto. Los criterios utilizados para el diseño de este robot son en su mayoría puramente técnicos, basados en la realización de los objetivos de la manera más económica y eficiente posible, aunque en algunas piezas también se han seguido criterios estéticos. El diseño de cualquier proyecto está muy condicionado por el método de fabricación, en nuestro caso, la impresión 3D limita muy poco el modelo pero si hay que tener en cuenta la resistencia del material utilizado para dimensionar las piezas más pequeñas. La impresión 3D también nos ha dado la oportunidad de fabricar las piezas de tal manera que encajen y se acoplen unas en otras de una manera precisa.

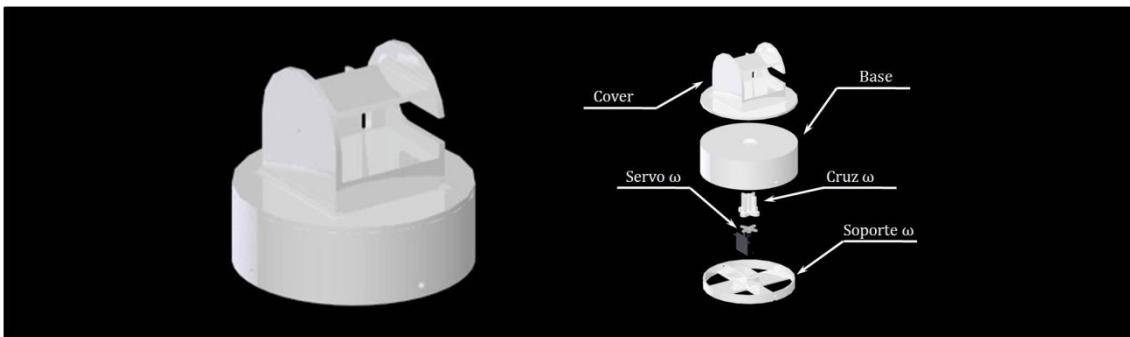


## Descripción física de piezas

El brazo robótico AURA está compuesto de 13 piezas impresas, 5 servos con sus 5 cruces y una placa controladora, absolutamente todas las piezas están impresas en ABS. Estructuramos el brazo en tres etapas para simplificar su explicación, estas etapas son la base, el brazo y la mano.

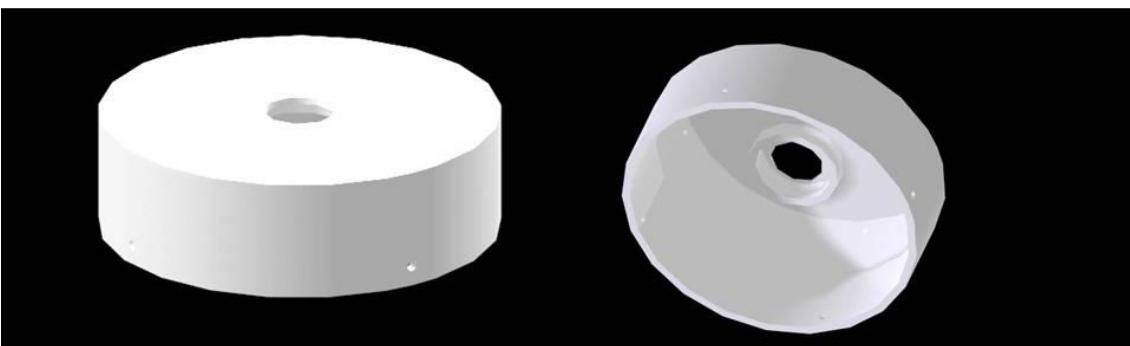


La primera etapa es la base, se encarga de dar estabilidad al brazo y del giro lateral. Está formada por cinco piezas: Base, servo omega, soporte omega, cruz omega y cover.



**Base**

La función de esta pieza es aguantar todo el peso del brazo y aportarle estabilidad, también alberga en su interior los mecanismos necesarios para el movimiento lateral del brazo. Tiene un diseño sencillo basado en dos parámetros importantes: el diámetro y el grosor de la pared. El diámetro ha de ser suficientemente grande como para aportar estabilidad sin quitar demasiada superficie de acción al brazo. La pared tiene que ser rígida y fuerte sin ser excesivamente gruesa. Tiene un agujero en su parte superior para conectar los mecanismos que alberga en su interior con el resto del brazo que tiene encima.



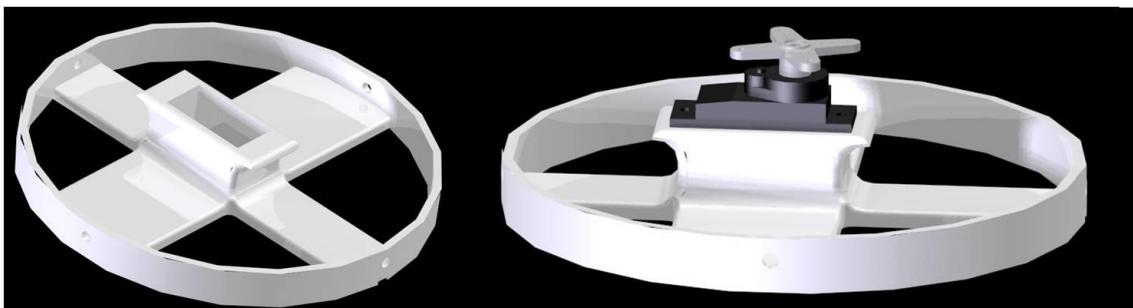
**Servo omega**

Es el servo encargado de la posición en la coordenada  $\omega$ , el giro sobre un eje vertical en el centro de la base. El servo elegido para esta función es el Turnigy TGY-R5180MG. Éste es un servo de 12g de peso, 2Kg·cm de fuerza de torsión, engranajes de metal y 180º hipotéticos de apertura de arco, que en la práctica se pueden extender. Este servo fue elegido por dos de sus cualidades principalmente: Tiene buena relación entre fuerza y tamaño, pero sobre todo, la apertura de arco por encima de los 180º es algo difícil de encontrar y vital para un brazo robótico. Este servo permite que el espacio de acción del brazo prácticamente se duplique respecto de otros modelos.



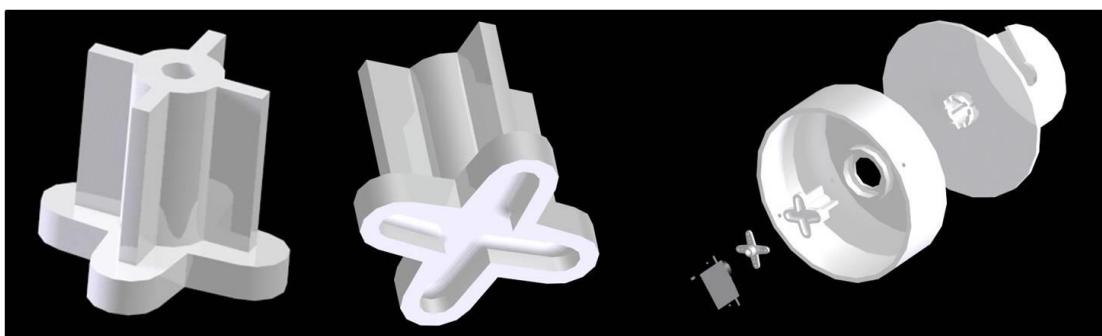
### Sopporte omega

Esta pieza tiene como única función albergar al servo omega y evitar que se mueva. Esta pieza encaja dentro de la base por lo que su diámetro exterior coincide con el interior de ésta. En el centro del soporte hay una caja donde se introduce el servo y el resto de la pieza sigue criterios puramente estéticos.



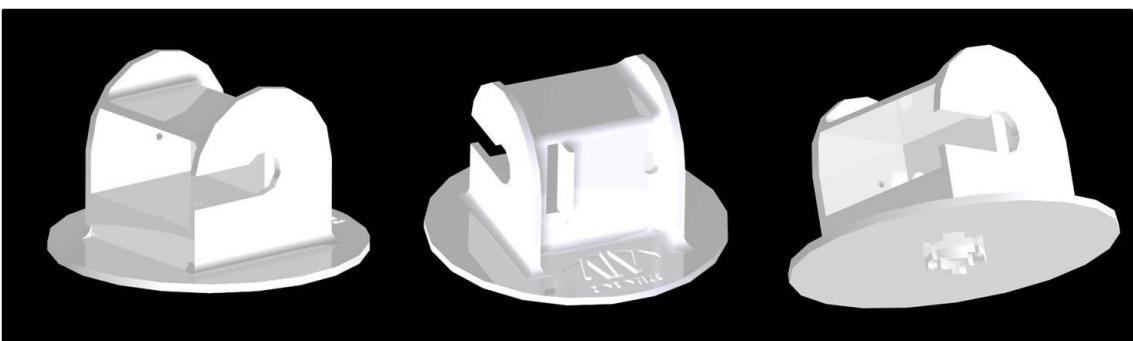
### Cruz omega

Es la pieza encargada de trasmitir el movimiento desde el servo, dentro de la base, hasta el cover. Tiene un negativo de la cruz propia del servo para encajar en ella y una forma de cruz extruida que encaja a través del agujero que posee la base hasta el negativo que tiene el cover.

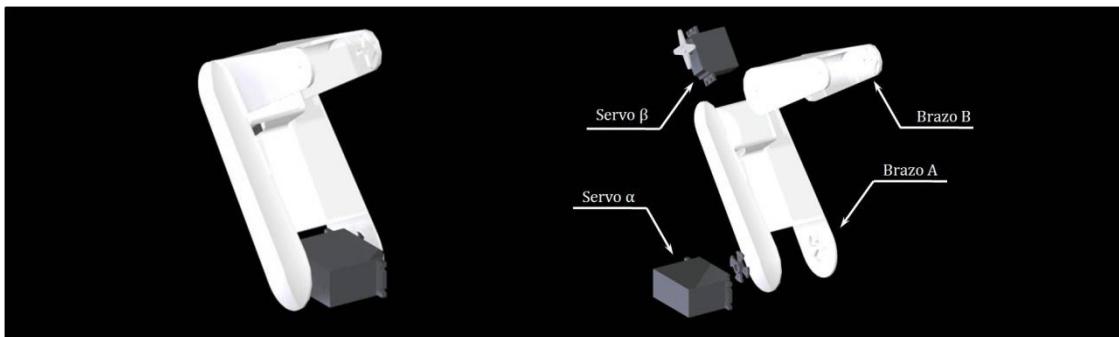


### Cover

El cover es una de las piezas claves del brazo robótico desempeñando tres funciones de alta importancia. El pequeño cilindro en la parte inferior de la pieza encaja en el agujero central de la base y posee un negativo en su interior por lo que encaja con la cruz omega, que sale por el mismo agujero, recibiendo así el movimiento lateral. La parte superior de la pieza está diseñada para enlazar con la segunda etapa, las formas semicirculares de la parte superior evitan los movimientos laterales del brazo respecto al cover. Y por último esta pieza alberga y sujetla el servo alfa, el más grande y potente que tiene este brazo. Este servo produce grandes fuerzas sobre la pieza por lo que su resistencia es vital. El logro del cover está en la realización de estas tres tareas en una única pieza y aportando acabados estéticos.

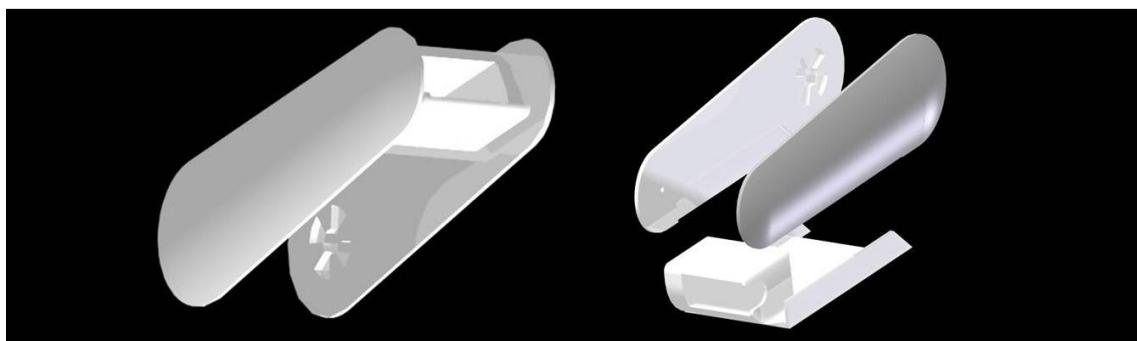


La segunda etapa es el brazo. Se encarga del movimiento en los ejes de la altura y la distancia y de aportar la fuerza necesaria para elevar y mover los objetos cogidos. Esta etapa se compone por dos brazos y los servos encargados de sus movimientos.



**Brazo A**

Es la pieza más grande del robot y una de las más complejas. Es un brazo en forma de U unido en su parte inferior al cover y en su superior al brazo B. El brazo está movido por el servo alfa, uno de los laterales del brazo posee un negativo de la cruz de este servo para que encaje y se eviten deslizamientos. La distancia entre los ejes de unión con el cover y con el brazo B es de 14 cm, este será el valor que usaremos para los cálculos de posición. Este brazo también alberga el soporte y la sujeción del servo beta en la parte superior. Esta pieza aporta soluciones estéticas a problemas estructurales como el aumento de grosor de las paredes mediante una superficie curva. Debido a su tamaño y complejidad, el brazo A esta impreso en tres piezas independientes y montado con posterioridad. Para el ensamblaje se ha utilizado adhesivo con base de cianoclorato.



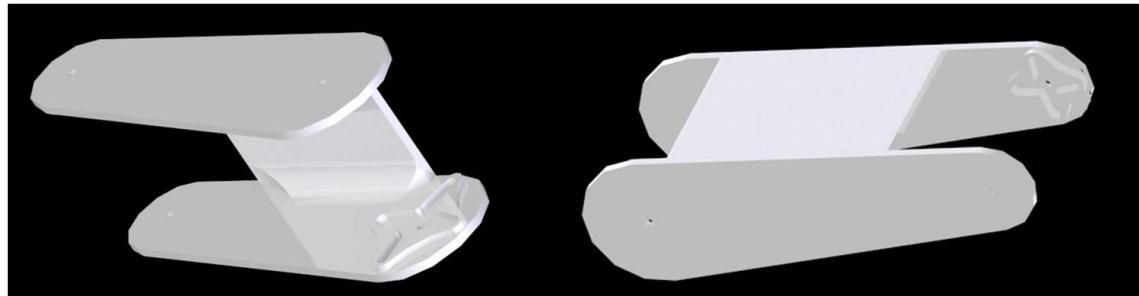
**Servo alfa**

Es el servo destinado a mover el brazo A y el encargado de levantar todo el peso del brazo robótico más el peso del objeto a mover, todo desde el extremo final de un brazo de palanca que puede llegar a 35cm. El servo VS-11 Vigor Extra Large fue elegido por su gran fuerza que nos evita los problemas añadidos de tener dos servos en esta posición. Las especificaciones de este servo son: 103g de peso, un máximo de 19Kg·cm de fuerza de torsión, una velocidad máxima de 0,19 segundos en hacer 60 grados y una amplitud máxima de 165º. El gran tamaño de este servo condicionará el ancho de todo el brazo robótico. La cruz de serie de este servo es redonda y por lo tanto ha de ser modificada con unos cortes para lograr una forma de cruz que encaje en el brazo A y evite los deslizamientos.



### Brazo B

Con una geometría más simple que el brazo anterior, mantiene un perfil en U y la impresión en tres piezas independientes pegadas después. Está unido en su parte inferior al brazo A y en su superior al brazo C, entre los ejes que lo unen con ambas piezas hay 12cm. El brazo B posee en la parte inferior de uno de sus laterales un negativo de la cruz del servo beta, para ayudar a trasmitir el movimiento y evitar deslizamientos. Lo mismo podemos decir la cruz del servo gamma, que también posee un negativo en la parte superior del lateral opuesto.

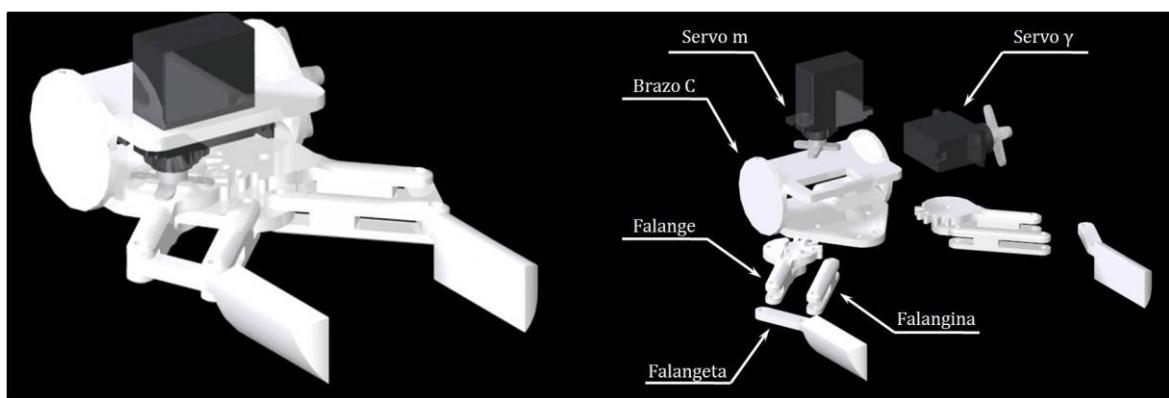


### Servo beta

Este servo da fuerza y movimiento al brazo B pero está situado en el extremo del brazo A, siendo el servo el propio eje de unión. En esta posición necesitamos un servo fuerte pero que no pese mucho. El servo HK15288A Analog BB/MG tiene 9 Kg·cm de fuerza de torsión, con 51g de peso, una velocidad de 0,21 segundos en hacer 60 grados y un arco de acción que podemos usar desde 30º hasta cerca de los 180º.

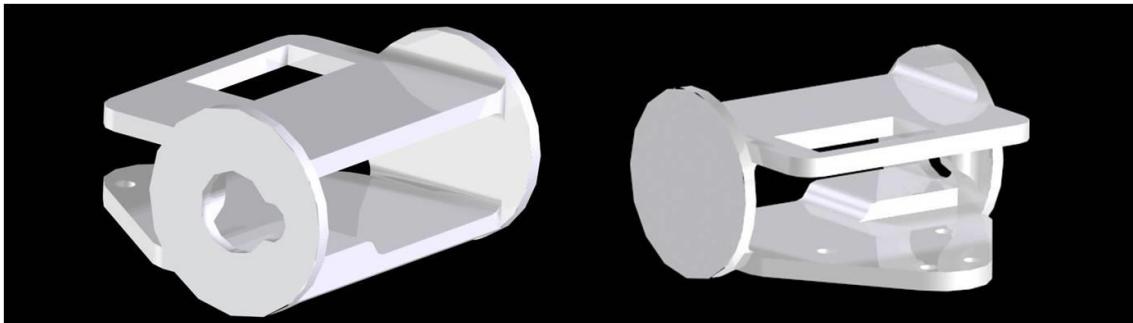


La tercera etapa es la mano, se encarga de agarrar los objetos y de la posición  $\theta$ , el ángulo de ataque respecto del suelo. Está compuesta por el brazo C, el servo gamma, el servo mano y dos dedos. Cada dedo se divide en una falange, una falangina y una falangeta.



### Brazo C

En el interior de este brazo encontramos dos servos y otras cuatro piezas, todo organizado para caber en el reducido espacio que permite esta pieza. El brazo C es la estructura y la base de toda la mano. Esta pieza está formada por cuatro planchas impresas individualmente, dos verticales con forma circular en los laterales y dos horizontales uniéndolas. En la plancha superior tenemos el corte donde encaja el servo mano y en uno de los laterales encontramos la estructura que sujeta el servo gamma, que es el encargado del movimiento de este brazo. En la plancha inferior están los cuatro agujeros donde se encajan las dos falanges y las dos falanginas.



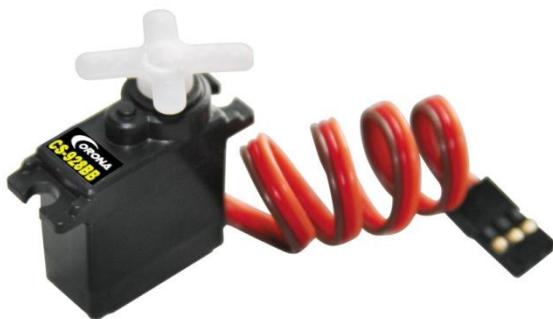
**Servo gamma**

Este servo es el encargado del movimiento de brazo C. En esta posición buscamos un servo muy ligero y con una apertura de arco lo mayor posible. Con estas características el mejor que encontramos es el servo Turnigy TGY-R5180MG, que es casualmente el mismo que utilizamos en el servo omega, dentro de la base. Recordamos las especificaciones: 12g de peso, 2Kg·cm de fuerza de torsión, engranajes de metal y un arco de acción que usaremos desde los 30º hasta los 300º. El servo gamma está colocado dentro del brazo C y su cruz encaja en el extremo del brazo B, siendo el propio servo la unión entre ambos brazos.

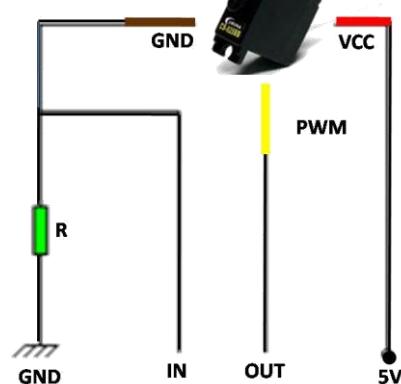


**Servo mano**

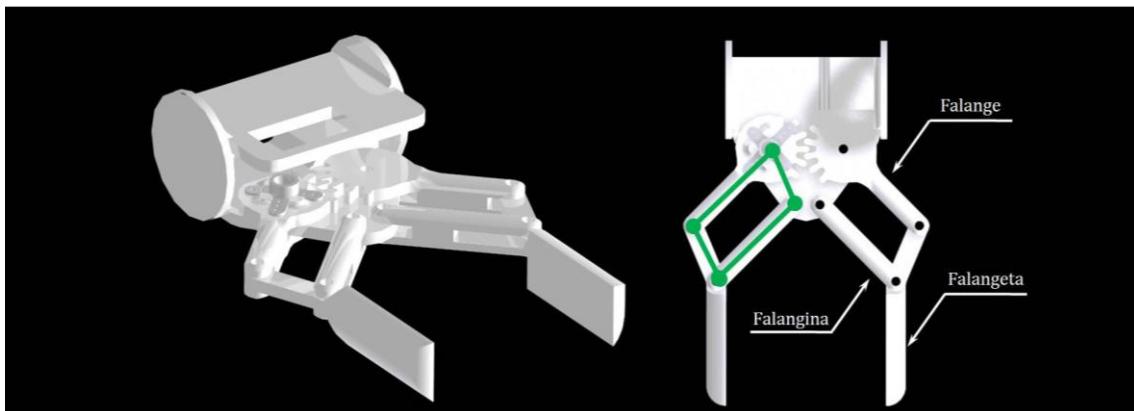
Este servo posiciona los dedos en el ángulo adecuado, no ha de ser muy fuerte pero sí muy ligero. Con solo 9g de peso, el servo Corona 928BB es el más adecuado para esta posición, tiene 2 Kg·cm de fuerza de torsión y una velocidad de 0,13 segundos en realizar 60 grados. Este servo está situado en la plancha superior del brazo C y su cruz está pegada a la falange del dedo derecho. Uno de los problemas que primero hay que plantearse en un brazo robótico es como saber cuándo parar de cerrar los dedos, no es lo mismo coger una botella o un bolígrafo. Muchos brazos de este tipo aportan soluciones parciales como prefijar manualmente el ancho del objeto a coger o apretar al máximo y poner un resorte para que el servo no se dañe. En el brazo robótico AURA se soluciona este problema monitorizando la fuerza ejercida por el servo. La fuerza que realiza un servo está directamente relacionada con su consumo, por lo tanto podemos utilizar una de las entradas analógicas de la placa controladora como voltímetro y cuando el dedo tope con el objeto el valor de esta entrada se elevará. Prefijamos la fuerza máxima que queremos que la mano realice sobre el objeto y cuando agarre a éste se parará.



Dedos

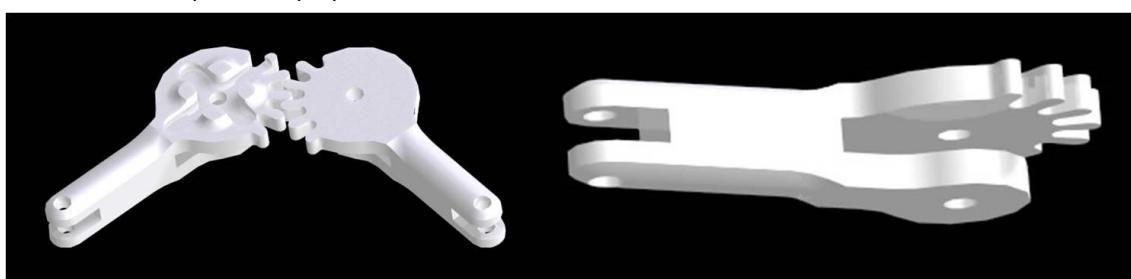


El brazo robótico tiene una mano compuesta por dos dedos en forma de pinza. Los dedos son simétricos y tienen un movimiento de paralelogramo, dos pares de lados paralelos que no varían en longitud pero sí el ángulo entre ellos. Uno de los detalles que diferencia a AURA de la gran mayoría de brazos robóticos caseros es el gran tamaño de su mano, con una apertura máxima de 68,4 mm que permite coger objetos tan cotidianos como una lata de refresco. Cada dedo está compuesto de:



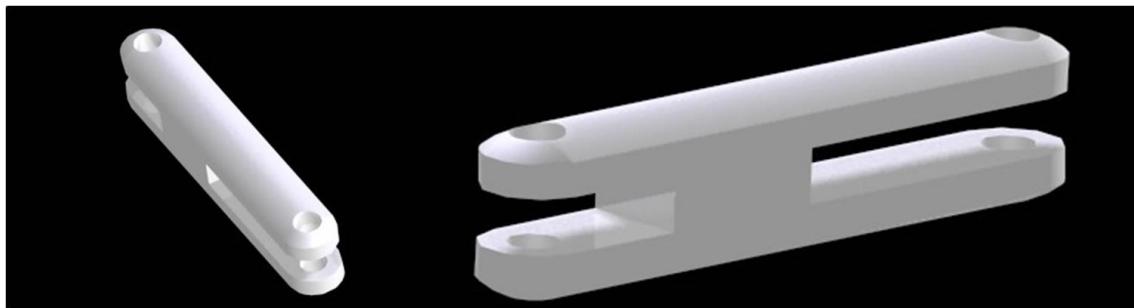
Falange

Es la pieza que transmite el movimiento y la posición al dedo. La falange derecha tiene un negativo de la cruz del servo mano, y por medio de una rueda dentada trasmite el movimiento a la otra falange que es simétrica a excepción del negativo de la cruz. De esta forma ambos dedos siempre mantienen el mismo ángulo de apertura. La pieza abraza en su parte posterior a la plancha inferior del brazo C, y a la falangeta en su parte anterior. La estructura de dos largueros unidos por el centro, aporta la rigidez necesaria en una pieza tan pequeña.



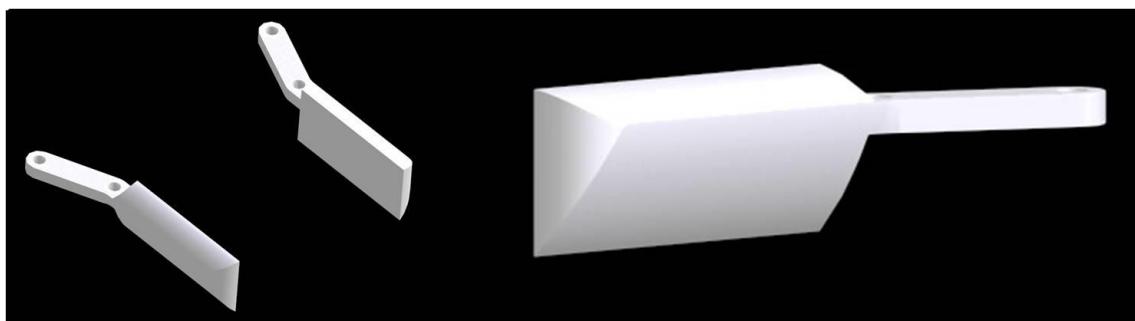
Falangina

Tiene la misma longitud que la falange y siempre se mueve paralela a ésta para conseguir un paralelogramo que varía en sus ángulos pero no en sus lados, este tipo de movimiento permite que ambos dedos estén siempre paralelos. Está diseñada con la misma estructura que la falange para aguantar el peso de los objetos. La falangina une el brazo C y la falangeta. Ambas falanginas son exactamente iguales.



**Falangeta**

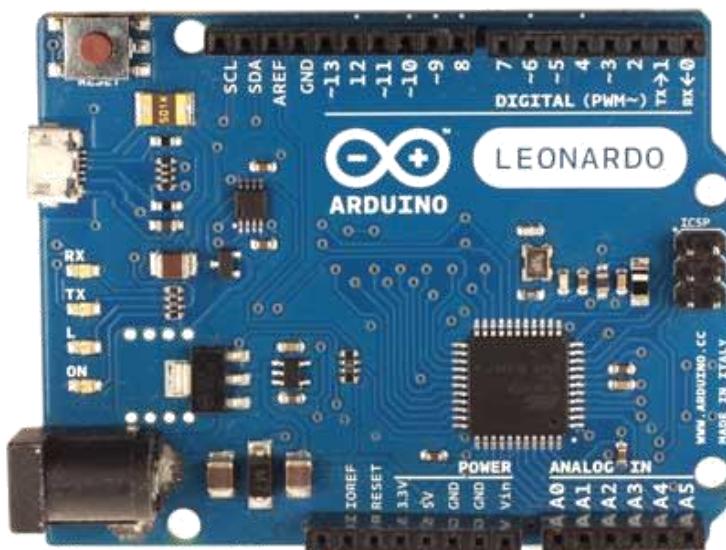
Es la última pieza del dedo y la destinada a coger los objetos. Está unida a la falange y a la falangina. Las dos falangetas son simétricas y se mueven de tal manera que siempre están paralelas. En esta pieza tenemos que destacar el aumento de superficie de contacto para agarrar mejor los objetos y el acabado estético de la cara exterior.



Hay otras dos piezas que no pueden englobarse en ninguna etapa pero son de vital importancia. Una es la placa controladora y la otra son los ejes de unión en cada articulación.

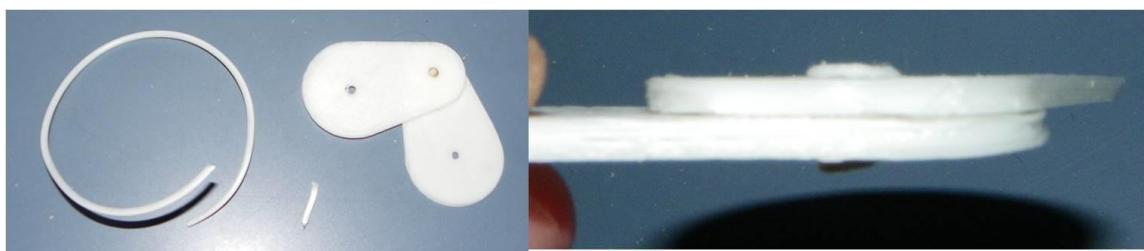
#### **Placa controladora**

Arduino es una plataforma abierta de electrónica con una gran variedad de productos. La placa controladora más económica que tiene las especificaciones adecuadas para este proyecto es el Arduino Leonardo. Tiene 20 puertos digitales, 12 entradas analógicas, 5voltios de voltaje operativo y 16MHz de velocidad de reloj, todo operado desde un microcontrolador ATmega32u4. Esta placa se programa en lenguaje C y tiene un puerto USB para comunicación en serie con el ordenador.



## Ejes

La gran mayoría de modelos y proyectos realizados con una impresora 3D resuelven las articulaciones con tronillos, ejes metálicos u otros elementos externos. En la construcción del brazo robótico AURA no se ha utilizado ninguno de esos elementos para los ejes. La impresora no tiene resolución suficiente para imprimir un eje lo bastante fino como para no crear rozamientos y suficientemente resistente para aguantar las fuerzas generadas, pero la materia prima de la impresora es un filamento de 3 mm de diámetro de plástico ABS. Este termoplástico funde a 230°C, lo cual lo hace muy manejable. Si tomamos un trozo de filamento, lo calentamos suficiente y lo hacemos rodar sobre una superficie plana, lo convertimos en un eje recto y uniforme. Cortamos la distancia que se necesite y quemamos ambas puntas para conseguir los topes. De este modo creamos ejes del mismo material que el resto del proyecto, sin ningún coste añadido y de fácil sustitución. El brazo robótico tiene 12 ejes, todos construidos de esta manera. Esta idea ha sido tomada y adaptada de un video que se encontró en internet: <http://www.youtube.com/watch?v=WqNxYHWpYko>

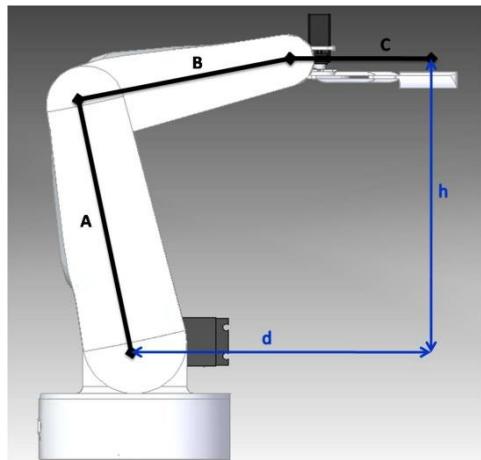
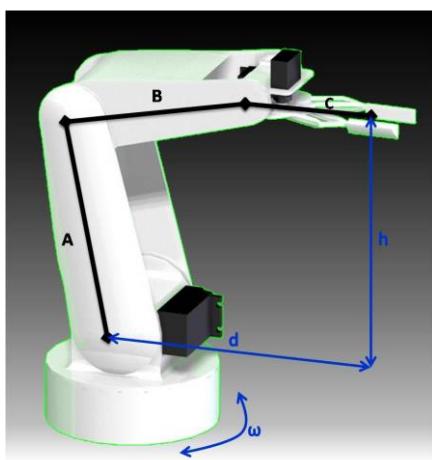


(Todos los planos necesarios para la definición y comprensión del diseño del brazo robótico están incluidos en los anexos a tamaño original)

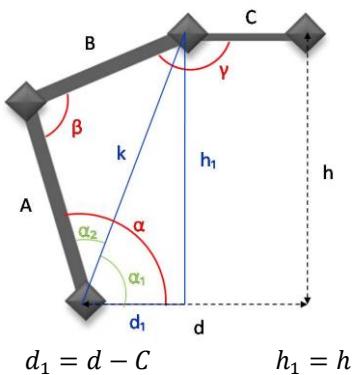
## Desarrollo matemático

### Cálculo de posición

Partimos del objetivo de controlar el brazo robótico por medio de unas coordenadas tridimensionales sin embargo tenemos que controlar los servos individualmente dándoles un ángulo donde posicionarse. Esto nos plantea un problema matemático de geometría. Como las coordenadas que vamos a utilizar son cilíndricas, una de ellas ( $\omega$ ) ya es un ángulo y por tanto no requiere ninguna operación matemática lo cual nos deja con un problema de geometría plana.



Como existe más de una combinación posible para que tres segmentos de distancia dada, uno a continuación de otro, lleguen de un punto a otro en un plano, necesitamos otro dato. Ese dato será el ángulo de incidencia del último brazo con el suelo ( $\theta$ ). Este dato podrá ser cambiado por el usuario a placer pero con el fin de dar estabilidad al objeto cogido, será por defecto cero, de esta manera la mano irá paralela al suelo. Esto nos facilita el cálculo matemático en los primeros pasos. Por lo tanto nos queda un pentágono de cinco lados conocidos y dos ángulos rectos. Dividimos el pentágono en un rectángulo y dos triángulos, uno de ellos rectángulo. Si aplicamos trigonometría:



Empezaremos con el ángulo  $\gamma$ . La suma de los ángulos de un polígono de "n" lados es:

$$S = 180^\circ(n - 2); \quad S = 180^\circ(5 - 2) = 540^\circ$$

Como tenemos un pentágono la suma de sus ángulos es  $540^\circ$ . Dos de sus ángulos son rectos, por lo tanto la suma de los otros tres será  $360$ .

$$\alpha + \beta + \gamma = 360; \quad \gamma = 360 - (\alpha + \beta)$$

Para hallar el ángulo  $\alpha$ , lo dividimos en dos ángulos  $\alpha_1$  y  $\alpha_2$  (ver gráfico). Hallaremos ambos ángulos por separado y después los sumaremos.

Usaremos la propia definición de tangente para hallar  $\alpha_1$  en el triángulo rectángulo de lados  $d_1$ ,  $h_1$  y  $k$ .

$$\tan \alpha_1 = \frac{h_1}{d_1}; \quad \alpha_1 = \tan^{-1} \left( \frac{h_1}{d_1} \right)$$

Para hallar  $\alpha_2$  usaremos el teorema del seno en el triángulo formado por A, B y k.

$$\frac{\sin \alpha_2}{B} = \frac{\sin \beta}{k}; \quad \alpha_2 = \sin^{-1} \left( \frac{B \sin \beta}{k} \right)$$

Ponemos k en función de  $d_1$  y  $h_1$  usando el teorema de Pitágoras en el triángulo de lados k,  $d_1$  y  $h_1$ .

$$d_1^2 + h_1^2 = k^2; \quad k = \sqrt{d_1^2 + h_1^2}$$

Sumamos ambos ángulos y sustituimos k.

$$\alpha = \tan^{-1} \left( \frac{h_1}{d_1} \right) + \sin^{-1} \left( \frac{B \sin \beta}{\sqrt{d_1^2 + h_1^2}} \right)$$

Para calcular el ángulo  $\beta$ , el último que nos queda, utilizamos el teorema del coseno y el teorema de Pitágoras:

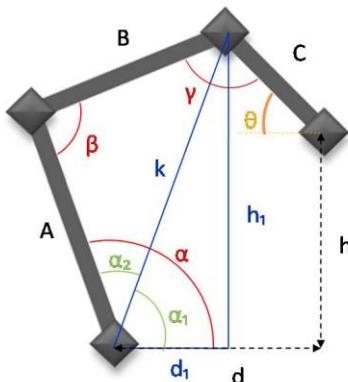
Primero nos centramos en el triángulo formado por los brazos A y B y la diagonal k. Si aplicamos el teorema del coseno al ángulo  $\beta$  en el triángulo mencionado obtenemos la fórmula:

$$A^2 + B^2 - 2AB \cos \beta = k^2$$

Sustituimos en la formula anterior el término  $k^2$  con el teorema de Pitágoras utilizado en el desarrollo del ángulo  $\alpha$ , obteniendo:

$$A^2 + B^2 - 2AB \cos \beta = d_1^2 + h_1^2; \quad \beta = \cos^{-1} \left( \frac{A^2 + B^2 - d_1^2 - h_1^2}{2AB} \right)$$

Con estas ecuaciones conseguimos los tres ángulos que necesitamos partiendo de los datos de la longitud de los brazos y las coordenadas pero recordemos que hay otro dato que el usuario puede



modificar, el ángulo de incidencia. Incluir  $\theta$  en las ecuaciones no es difícil.

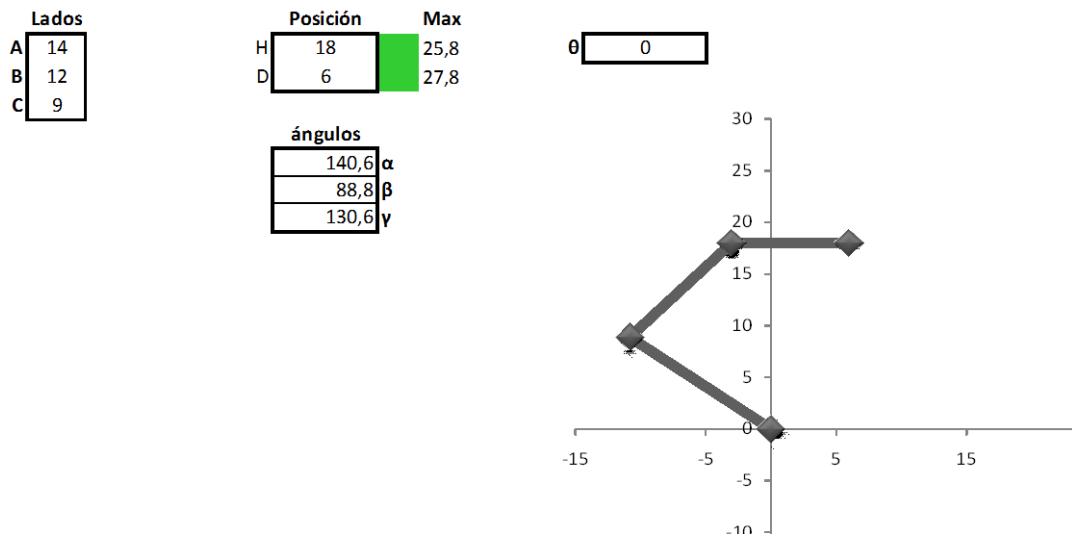
$$d_1 = d - C * \cos \theta \quad h_1 = h + C * \sin \theta \quad \gamma = 360 - (\alpha + \beta + \theta)$$

Cuando se trabaja con funciones trigonométricas hay más de una solución posible y las matemáticas no pueden distinguir cuál de los ángulos es el más conveniente para un brazo robótico. El programa informático elige por defecto la solución positiva más cercana al cero y en el caso de los ángulos  $\beta$  y  $\gamma$  esta selección es la acertada para nuestros propósitos pero en el caso del ángulo  $\alpha$  tenemos que incluir en el programa formulas lógicas condicionales en la sección del cálculo.

$$SI (d_1 = 0) \Rightarrow \alpha_1 = 90 \quad SI (\alpha < 0) \Rightarrow \alpha = \alpha_1 + \alpha_2 + 180$$

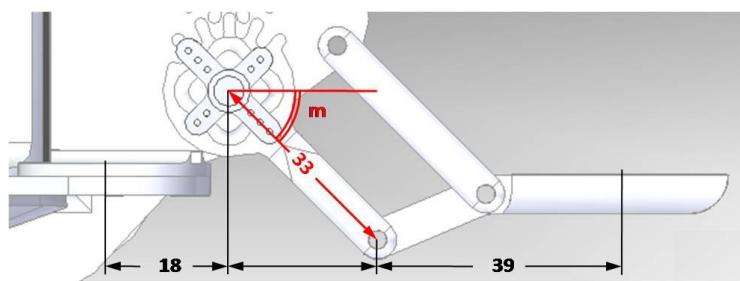
Si introducimos todas estas fórmulas y los datos necesarios en una hoja de cálculo y creamos una gráfica podemos hacer una representación esquemática del brazo robótico y simular sus distintas posiciones en el plano.

(La versión interactiva de esta hoja de cálculo será incluida en la página web)



El brazo A y el brazo B siempre mantienen la misma longitud, pero debido al sistema de apertura de la mano la distancia entre el eje que une el brazo B y el brazo C y los dedos varía según el ángulo de apertura de la mano. Si queremos abrir la mano sin variar la posición en el espacio tendremos que corregir la distancia que mide el brazo C en función de este ángulo. Podemos dividir esta medida en tres partes. La distancia desde el eje que une este brazo con el anterior hasta la unión de las falanges no varía. Luego tenemos el coseno del ángulo de apertura multiplicado por la longitud de la falange. Y por último la longitud de la falangeta, la última pieza del dedo, que para evitar que se resbalen los objetos consideraremos hasta la mitad de la pieza. El brazo C medirá la suma de estos tres valores en milímetros.

$$C = 18 + 33 * \cos(m) + 39$$



## Calculo de trayectoria

Conocemos la posición actual y la posición a la que queremos ir, buscamos la trayectoria más corta para unir ambos puntos y encontramos una recta. Si el movimiento es vertical o frontal no habrá ninguna complicación, simplemente iremos variando su posición a intervalos determinados para conseguir una velocidad adecuada, pero si queremos hacer un movimiento lateral nos damos cuenta de que la coordenada  $\omega$  es circular y el movimiento natural sería un arco. Si queremos buscar la trayectoria más corta tendremos que compensar el movimiento circular con una variación en la coordenada d.

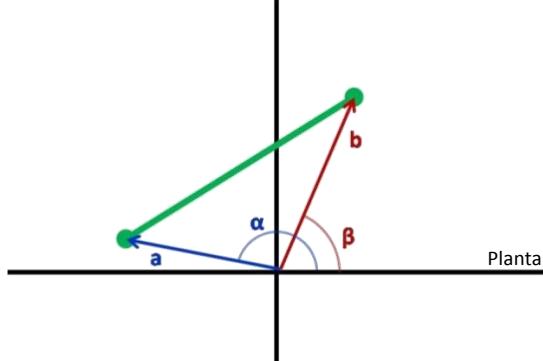
Para hallar esta recta en un sistema de coordenadas polares tenemos que partir de la ecuación de una recta que pasa por dos puntos y sustituir las coordenadas cartesianas por equivalencias polares. Si queremos ir de un punto de distancia a y ángulo  $\alpha$  a un punto de distancia b y ángulo  $\beta$ , y entendemos que  $\omega$  es el ángulo que vamos a ir variando, entonces la ecuación nos queda:

$$\frac{x - x_a}{x_b - x_a} = \frac{y - y_a}{y_b - y_a} \frac{d * \cos(\omega) - a * \cos(\alpha)}{b * \cos(\beta) - a * \cos(\alpha)} = \frac{d * \sin(\omega) - a * \sin(\alpha)}{b * \sin(\beta) - a * \sin(\alpha)}$$

En esta ecuación tenemos  $\omega$  como variable independiente, el ángulo que va tomando, y d como variable dependiente, la distancia que debemos tener en cada ángulo para que la trayectoria sea una recta. Despejamos d y simplificamos para obtener:

(1)

$$d(\omega) = \frac{a * b * \sin(\alpha - \beta)}{a * \sin(\alpha - \omega) - b * \sin(\beta - \omega)}$$



Sin embargo no siempre la trayectoria es tan simple como una recta. Tenemos que tener en cuenta que hay ciertas posiciones en la que no podemos colocarnos, ya sea por límites de ángulos de los servos o por colisión con la base. Esto hace que en las capas inferiores tengamos una zona circular a la que no podemos acceder. Si la recta que une el punto de salida y él de llegada es secante a este círculo tendremos que hallar otra trayectoria que no pase por esta zona. La trayectoria más corta para unir dos puntos con una zona prohibida circular será la unión de las dos rectas tangentes al círculo que pasen por los puntos de salida y llegada respectivamente y el arco de circunferencia entre ambos puntos de tangencia. Usamos la ecuación anterior para las rectas tangentes, sustituyendo el punto de llegada o salida por el punto de tangencia correspondiente. De esta manera nos queda una ecuación a trozos, que simplificada queda:

$$(2) \quad d(\omega) = \begin{cases} \frac{a * m * \sqrt{1 - \frac{m^2}{a^2}}}{a * \sin(\alpha - \omega) - m * \sin(\alpha - \cos^{-1} \frac{m}{a} - \omega)} & \forall \omega < (\alpha - \cos^{-1} \frac{m}{a}) \\ m & (\beta + \cos^{-1} \frac{m}{b}) \geq \omega \geq (\alpha - \cos^{-1} \frac{m}{a}) \end{cases}$$

$$\begin{cases} & \\ & \end{cases}$$

$$\begin{cases} \frac{b * m * \sqrt{1 - \frac{m^2}{b^2}}}{m * \sin(\beta + \cos^{-1} \frac{m}{b} - \omega) - b * \sin(\beta - \omega)} & \forall \omega > (\beta + \cos^{-1} \frac{m}{b}) \end{cases}$$



La zona a la que no se puede acceder está causada por los ángulos límites de los servos. Para averiguar el radio de este círculo tenemos que hallar la distancia mínima que permite cada servo en esas condiciones de altura, longitud del brazo C y ángulo de incidencia ( $\theta$ ). Como en el posicionamiento en un plano vertical actúan tres servos, tendremos tres ecuaciones y utilizaremos la que resulte más restrictiva al sustituir los datos conocidos.

$$m_\beta = \sqrt{A^2 + B^2 - AB\sqrt{3} + C \cos(\theta)}$$

$$m_\alpha = B \cos(\beta - 15^\circ) - \frac{(1 + \sqrt{3})A}{2\sqrt{2}} + C \cos(\theta)$$

$$m_\gamma = \sqrt{2} * (A \sin(105^\circ - \theta - \beta) - B \sin(105^\circ - \theta) - C \sin(\theta - 45^\circ))$$

Para saber de una manera matemática cuál de las dos fórmulas debemos usar en cada caso, tenemos que saber si la trayectoria directa es secante o exterior al círculo prohibido, en otras palabras, si la distancia mínima desde la recta hasta el centro de coordenadas es mayor o menor al radio del círculo. La distancia mínima se obtiene mediante la fórmula de distancia entre un punto y una recta.

$$d_{min} = \left| \frac{Ax + by + c}{\sqrt{A^2 + B^2}} \right| ; \quad d_{min} = \left| \frac{\mathbf{a} * \mathbf{b} * \sin(\beta - \alpha)}{\sqrt{\mathbf{a}^2 + \mathbf{b}^2 - 2 * \mathbf{a} * \mathbf{b} * \cos(\beta - \alpha)}} \right|$$

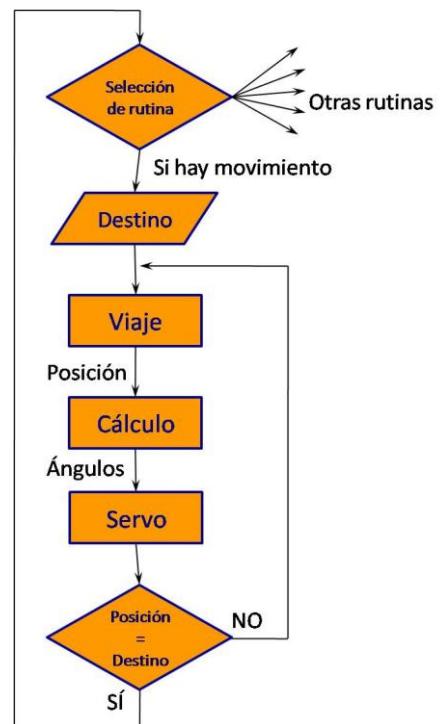
$$\text{si } d_{min} \geq m \Rightarrow (1) \quad \text{si } d_{min} < m \Rightarrow (2)$$

Una vez tenemos calculadas las trayectorias simples en las distintas coordenadas no es complicado obtener una trayectoria combinada en distintos ejes mediante composición de movimientos. Si queremos ir desde un punto A ( $\omega_a, d_a, h_a, \theta_a$ ) hasta un punto B ( $\omega_b, d_b, h_b, \theta_b$ ) tendremos tres movimientos simultáneos, por una parte iremos variando la altura, desde  $h_a$  hasta  $h_b$ , a una velocidad que permita empezar y acabar todos los movimientos al mismo tiempo, por otra parte haremos lo mismo en la coordenada  $\theta$  de forma simultánea y finalmente las coordenadas  $\omega$  y  $d$  están relacionadas en una única expresión. En caso de ser la ecuación por partes (2) tendremos que tener en cuenta la variación del radio mínimo en función de  $h$  y  $\theta$ . Para conseguir que todos los movimientos comiencen y finalicen al mismo tiempo debemos hallar la velocidad que han de tener cada uno de ellos para que tarden el mismo tiempo en recorrer distintas distancias. Primero dividimos la distancia que debe recorrer cada uno entre la velocidad máxima estipulada en ese eje, con eso hallamos el tiempo mínimo de cada movimiento por separado. Tomamos el mayor de los tiempos y dividimos cada distancia entre este valor, obteniendo la velocidad necesaria en cada movimiento para que todos duren lo mismo. De esta manera obtenemos la trayectoria más corta y el movimiento más natural.



## Programación

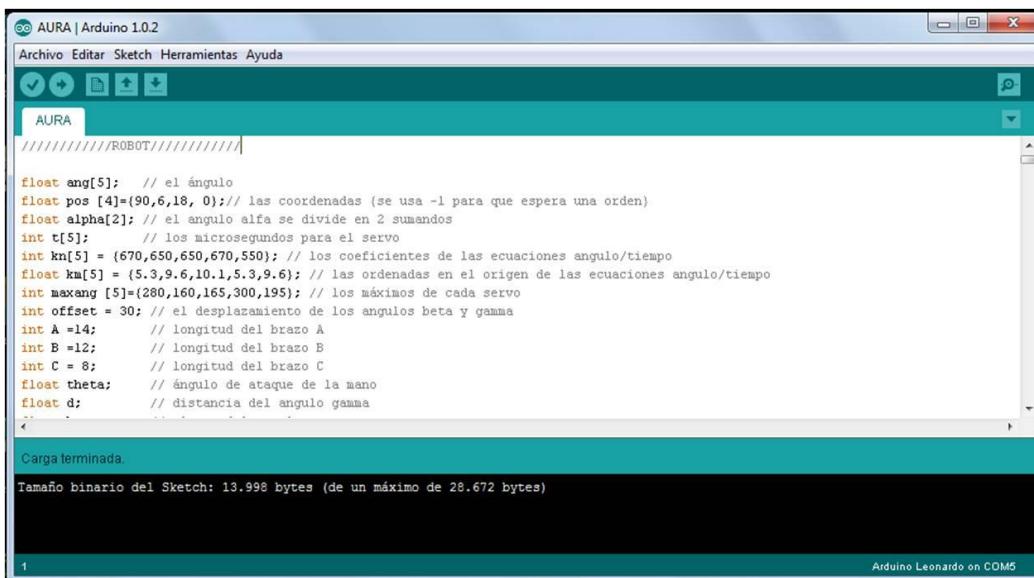
Hay muchas maneras de controlar un brazo robótico de tipo industrial. La gran mayoría de brazos robóticos industriales son controlados por una sucesión de comandos directamente a los servos que les indican el ángulo que deben tomar en cada momento y repiten una secuencia de posiciones según convenga. Este modelo sería el utilizado en el clásico brazo de cadena de montaje que coge un objeto de una cinta y lo coloca o clasifica en otro lugar. Si buscamos que nuestro brazo este enfocado al usuario en lugar de a un único uso determinado, tendremos que aportar alguna entrada y reaccionar en consecuencia. La primera forma que debemos plantearnos es el control directo, donde el usuario introduce el ángulo donde debe estar cada servo. Esta es la forma de control por usuario más extendida con mucha diferencia, aunque no siempre la entrada es directa por parte del usuario, generalmente se suele usar un interfaz como un joystick, un ratón, o un mando de consola. Este método es simple pero no es intuitivo para el usuario, por ello en AURA incluimos una sección de cálculo matemático que permite obtener los ángulos desde unas coordenadas en el espacio mucho más intuitivas. De esta manera introducimos la posición a la que queremos llevar el brazo y automáticamente se posiciona allí.



El principal problema de la programación de un brazo robótico es la posición que deben tener los servos para llegar a un punto concreto en el espacio, pero no es el único. Uno de los problemas más olvidados en la programación son los movimientos, ninguno de los brazos robóticos de este tipo se preocupa por tener unos movimientos optimizados, simultáneos y naturales. Una de las secciones más importantes en la programación del brazo robótico AURA son los viajes, en esta sección se controla la velocidad, la aceleración y la trayectoria de los movimientos. Para ello usamos el cálculo de trayectoria que nos dice cuál será el próximo paso y avanzamos distancias pequeñas a intervalos de tiempo calculados para controlar la velocidad y algo aún más importante, la aceleración. Si buscamos cambios de velocidad progresivos en vez de arranques y paradas bruscas evitaremos que se nos caigan los objetos, sobre todo en los comienzos y en los finales de los movimientos.

Para controlar el brazo mandamos comandos por comunicación en serie hasta la placa controladora, allí se interpretan. La primera parte del programa recibe ese comando y decide la rutina que debe lanzar. Para simplificar el análisis y la comparación de los comandos se usan caracteres, de esta manera si queremos movernos hacia la derecha enviaremos una "r" o enviaremos una "g" si queremos ir a un punto concreto en el espacio. El programa usa un carácter para cada rutina pero gracias al interfaz el usuario no tiene que conocer los comandos para controlar el brazo. El programa controla al brazo, interpreta la entrada, calcula las trayectorias, maneja los servos, calcula sus posiciones y otras muchas funciones secundarias. Una de estas funciones es proteger a los servos en caso de que se intente ir a un ángulo fuera de su rango, o cuando intenta cargar más peso del que puede levantar.

El programa que se carga en la placa controladora está escrito en lenguaje de programación C, en un entorno facilitado por Arduino. El lenguaje C es uno de los más usados y reconocidos en todo el mundo, es un lenguaje de nivel medio que se caracteriza por sus muchas bibliotecas y variantes del lenguaje, en nuestro caso usamos RobotC, una variante con rutinas especializadas para placas controladoras periféricas, es decir fuera de un ordenador. El entorno de Arduino es totalmente gratuito y muy cómodo para comunicarte con una placa de este estilo.



```

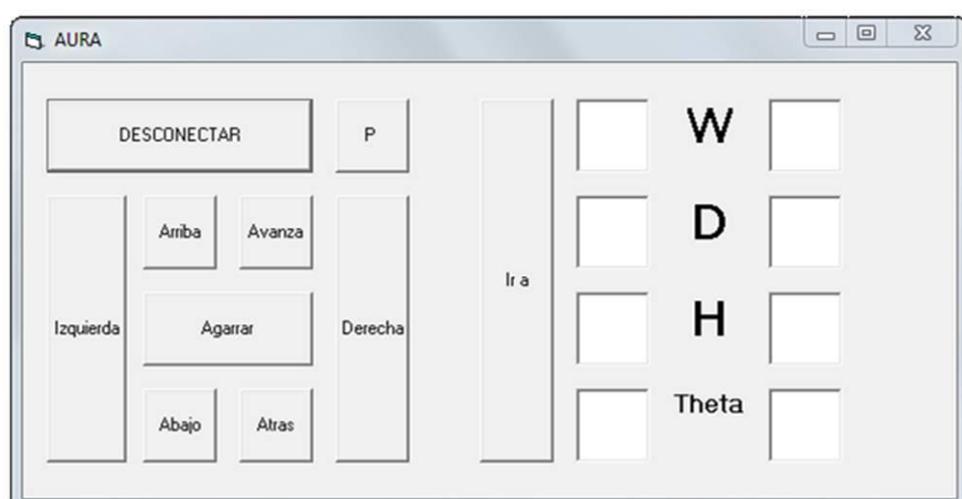
// AURA | Arduino 1.0.2
Archivo Editar Sketch Herramientas Ayuda
AURA
//////////ROBOT///////////
float ang[5]; // el ángulo
float pos [4]={90,6,18, 0}// las coordenadas (se usa -l para que espera una orden)
float alpha[2]; // el angulo alfa se divide en 2 sumandos
int t[5]; // los microsegundos para el servo
int kn[5] = {670,650,650,670,550}; // los coeficientes de las ecuaciones angulo/tiempo
float km[5] = {5.3,9.6,10.1,5.3,9.6}; // las ordenadas en el origen de las ecuaciones angulo/tiempo
int maxang [5]={280,160,165,300,195}; // los máximos de cada servo
int offset = 30; // el desplazamiento de los angulos beta y gamma
int A =14; // longitud del brazo A
int B =12; // longitud del brazo B
int C = 8; // longitud del brazo C
float theta; // ángulo de ataque de la mano
float d; // distancia del angulo gamma
...
Carga terminada.
Tamaño binario del Sketch: 13.998 bytes (de un máximo de 28.672 bytes)

1
Arduino Leonardo on COM5

```

## Interfaz y control

Podríamos definir un interfaz como un medio que usamos para controlar una máquina o un programa, por ejemplo un teclado, un joystick u otro programa. En este caso utilizaremos otro programa diferente al que usamos para controlar el brazo robótico. Este programa es un interfaz gráfico, es decir una ventana dónde utilizamos botones para enviar comandos al robot. El programa que utilizaremos como interfaz está programado en un entorno diferente al programa controlador, esta vez utilizamos el entorno Visual Basic. Éste es un entorno del grupo Microsoft enfocado a programas sencillos para ejecutar dentro del sistema operativo Windows. Las grandes ventajas que nos aporta un entorno como éste en el proyecto son la facilidad para programar elementos gráficos y ventanas, y la posibilidad de utilizar el puerto serial de Windows sin necesidad de terceros programas.

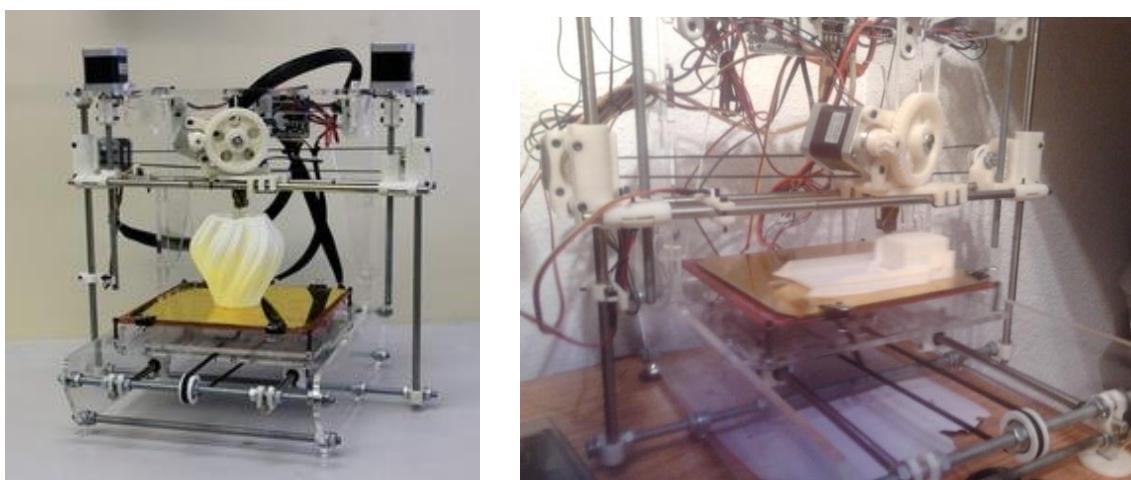


Finalmente el proceso de control quedaría de la siguiente manera: Pulsamos un botón en la ventana del interfaz, el programa envía por el puerto serial un carácter asociado a este botón, la placa controladora recibe este carácter y activa una rutina, la placa devuelve información que puede ser útil al usuario, el interfaz muestra esta información y el ciclo vuelve a empezar.

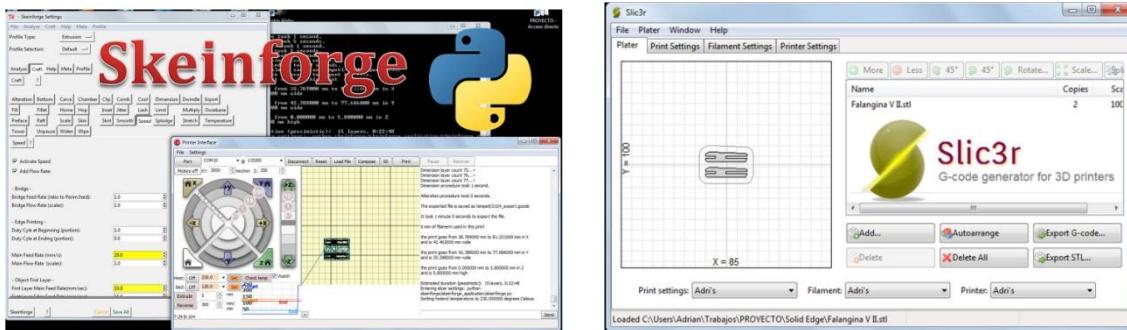
El usuario tiene dos maneras de mover el brazo robótico, mediante órdenes y mediante viajes. Una orden es un comando de movimiento directo, como “sube”, “baja”, “cierra la pinza” o “ve a la posición inicial”. Los botones que corresponden al control por órdenes están en la parte izquierda de la ventana, y a la derecha encontramos los botones para el control por viajes. Un viaje se da cuando mandamos al brazo posicionarse en un lugar en el espacio concreto, alejado de su posición actual. Para controlar el destino de estos viajes tenemos cuatro entradas de código para introducir las coordenadas de este punto y un botón para comenzar el movimiento. Cuatro salidas de datos nos indican las respectivas coordenadas de la posición encada instante.

## Impresión 3D

La impresión 3D es un grupo de métodos de fabricación basados en el control numérico que mediante la superposición de capas bidimensionales de material crea un objeto tridimensional desde un modelo virtual. Hay muchos métodos de impresión 3D pero el utilizado en la fabricación de AURA es el modelado por deposición fundida. La impresora utiliza como materia prima un polímero termoplástico, ABS en nuestro caso, lo funde y lo deposita por capas hasta construir el modelo. Este tipo de impresión es la más utilizada por el reducido coste que conlleva, pero incluso dentro del mismo método encontramos una gran variedad de impresoras. La que se ha utilizado en la creación del brazo robótico es la Prusa Air 2. Esta impresora es un modelo de bajo presupuesto de tipo experimental y con código abierto lo que significa que hay que conseguir las piezas, un gran número de ellas son impresas, y montarla por uno mismo sin una marca que te dé instrucciones o garantías. Esto hace que se abarate mucho su coste.



Para crear el brazo robótico AURA fue preciso diseñar cada pieza por separado en un entorno CAD, el programa elegido fue Solid Edge. Después para conseguir imprimir cada pieza fue necesaria una serie de transformaciones y cambios de formato. Primero había que transformar el archivo Solid Edge en un formato de archivo tridimensional abierto y universal, el más adecuado es el conocido como litografía o “.stl”. La impresora trabaja por control numérico por lo tanto es necesario transformar el archivo tridimensional en una serie de órdenes y posiciones en el espacio donde depositar el material para crear la pieza, el archivo que contiene estas órdenes recibe el nombre de Gcode. Para transformar un archivo tridimensional en un archivo Gcode hace falta un programa laminador donde introduces el archivo STL y una serie de opciones y ajustes específicos de cada pieza, como densidad, velocidad de impresión, temperatura, altura de capa y otros muchos parámetros. En la impresión de las piezas del brazo robótico se han utilizado dos programas laminadores distintos, Slic3r y Skeinforge. Estos dos programas son gratuitos y abiertos. Skeinforge es un programa desarrollado para impresoras industriales, es de gran fiabilidad y precisión pero tiene demasiados parámetros a controlar y unido a que es un programa muy poco intuitivo hace que sea muy difícil de manejar, además es considerablemente más lento que el Slic3r. Con características casi opuestas, Slic3r es un programa más novedoso, diseñado para usuarios no profesionales, es mucho más sencillo e intuitivo pero comete errores con más frecuencia, sobre todo en las piezas más difíciles.



La impresión 3D aporta una nueva perspectiva a proyectos como este brazo robótico. Para empezar, el hecho de disponer de un método de control numérico, como la impresión 3D, permite diseñar un proyecto en el ordenador y obtener el modelo con mucha precisión. Cualquier alternativa supone que aunque se diseñe en un entorno CAD, al construirse se perderá toda la precisión que aporta el ordenador y estará limitado por el material y el método de fabricación que se use. Con una impresora 3D los límites de diseño desaparecen, se puede diseñar prácticamente cualquier pieza sin preocuparse por la construcción. A todas estas ventajas tenemos que sumarle el reducido coste de la materia prima, que abarata el coste final y permite imprimir una pieza para ver su aspecto real, corregir los fallos y volverla a imprimir tantas veces como se necesite. Esto unido al poder generar recambios totalmente idénticos en caso de rotura demuestra la importancia de poder repetir la fabricación de una pieza siempre con la misma precisión.



## Memoria

Desde la misma elección de un proyecto hasta las pruebas finales de su funcionamiento, las ideas que se van teniendo sobre el acabado y las soluciones que se van dando a los problemas que surgen cambian progresivamente. En esta sección se hace un repaso a todos los cambios de concepción que ha sufrido el brazo robótico a lo largo del tiempo.

Como todo proyecto, éste se comenzó con una gran búsqueda de información para ver que podía aportar AURA respecto al resto de brazos que ya existían. Los primeros procesos que se abrieron fueron el diseño, primero con bocetos a lápiz y más tarde en ordenador; el cálculo matemático, desde el principio se manejó la idea del uso de coordenadas en lugar del control directo; y la selección de material. Se decidió que sería un proyecto de presupuesto muy reducido, se buscaba trabajar con materiales baratos, fáciles de obtener y moldear. Haciendo un gran trabajo de documentación encontramos una larga lista de materiales válidos, pero el más adecuado era el PVC. Este plástico puede obtenerse en forma de tubería en cualquier tienda de fontanería por un coste muy reducido. Además el PVC es un termoplástico que funde cerca de los 140°C, por lo que podía fundirse en un horno de casa y modelarse con facilidad. Se eligió este material, se empezó a hacer pruebas y se comenzó el diseño. En este punto se eligieron los cinco servos y se encargaron. Pero el PVC traía complicaciones constantes, el grosor mínimo que resistía las pruebas de resistencia era de 5mm y las piezas con este grosor eran demasiado pesadas. El mayor problema que surgió fue el método de construcción: se fundía una tubería con un corte longitudinal y se abría para crear una plancha, en esta plancha se recortaban las piezas desplegadas y con calor local se doblaban para formar la pieza deseada. Este método era barato y sencillo pero limitaba mucho el diseño, ya que todas las piezas tenían que provenir de planchas dobladas, y era demasiado inexacto, toda la precisión que aportaba el diseño por ordenador se perdía en planchas mal alisadas y dobleces poco precisos. Esto junto a otros problemas obligó a descartar el PVC con un tercio del tiempo ya perdido y con un diseño que habría que adaptar.



Se volvieron a barajar varias opciones, y destacó una alternativa similar: Las planchas de metacrilato cortadas a laser de forma profesional. Esta opción reducía el peso y el grosor de las piezas y permitía un método de construcción por control numérico, existen empresas que desde un fichero CAD cortan las piezas a medida. Esta opción traía muchas ventajas pero su coste era demasiado alto. A falta de una solución mejor se adaptó el diseño al nuevo material pero muy poco después surgió otra posibilidad que empezaba a desarrollarse, la impresión 3D.

La impresión 3D ya existía desde hacía años pero el coste de la propia máquina lo descartaba como opción desde el principio. Sin embargo empezó a existir un movimiento de impresoras 3D abiertas y de bajo coste en pequeños círculos de gente cercana a las universidades e involucrada en el desarrollo de esta tecnología. Leap to 3D es una pequeña empresa que intenta expandir el uso de estas impresoras, gracias a ellos la opción de la impresión 3D se convirtió en una solución asequible.

Se obtuvieron las piezas para montar una impresora Prusa Air 2, se ensambló, se consiguieron los programas, se calibró y se aprendió a usarla. Se perdió mucho tiempo en este proceso pero permitió un cambio de mentalidad radical en el proyecto: Las piezas serían más exactas y más ligeras, el diseño no tendría que partir desde una plancha, el precio del robot se reduciría respecto del PVC y otras muchas mejoras.

El brazo se tuvo que rediseñar por cuarta vez, y se empezaron a imprimir las piezas. El nuevo método de fabricación había solucionado todos los problemas anteriores pero trajo unos pocos nuevos. Cuando hubo que imprimir los brazos surgieron imperfecciones que producía la propia impresora en piezas largas y altas, para solucionar este problema se decidió dividir los brazos en tres piezas de menor tamaño y pegarlas después. Para incluir estos cambios y solucionar problemas menores de diseño se hizo la última versión del brazo robótico, que sería la definitiva.



## Presupuesto

| Concepto                      | Precio  | Cantidad | Coste          |
|-------------------------------|---------|----------|----------------|
| Módulo Arduino Leonardo       | 14,98\$ | 1        | 10,88€         |
| Servo Corona 928BB            | 2,99\$  | 1        | 2,17€          |
| Servo HK15288A Analog BB/MG   | 6,11\$  | 1        | 4,44€          |
| Servo VS-11 Vigor Extra Large | 9,79\$  | 1        | 7,11€          |
| Servo Turnigy TGY-R5180MG     | 4,99\$  | 2        | 7,25€          |
| Filamento ABS natural 3mm     | 22€/Kg  | 0,36 Kg  | 7,98 €         |
| <b>COSTE TOTAL</b>            |         |          | <b>39,83 €</b> |

(Conversiones de dólares a euros hechas el 16/12/2013)

El presupuesto en este proyecto es una sección que ha tomado mucha importancia. Desde el primer momento se intentó reducir al mínimo el coste del brazo, aunque eso significara perder algo de calidad en los servos. Para abaratar el proyecto todos los artículos han sido comprados por internet a mayoristas internacionales especializados en electrónica y modelismo, entre ellos "hobbyking" y "dealextreme". El filamento ha sido comprado a la misma empresa donde obtuvimos la impresora, "leap to 3d".

## Bibliografía

### Documentación

<http://es.slideshare.net/elvisrichard/brazo-robtico-1775457>  
[http://robots-argentina.com.ar/Robots\\_UnBrazoRobotico.htm](http://robots-argentina.com.ar/Robots_UnBrazoRobotico.htm)  
[http://www.societyofrobots.com/robot\\_arm\\_tutorial.shtml](http://www.societyofrobots.com/robot_arm_tutorial.shtml)  
<http://es.scribd.com/doc/72216305/Brazo-Robotico>  
<http://jishortcut.wordpress.com/2010/04/19/my-mini-servo-grippers-and-completed-robotic-arm/>  
<http://www.thingiverse.com/thing:2433>  
<http://jishortcut.wordpress.com/2010/08/17/multiply-the-robotic-arm-and-electronics/>  
[http://people.ece.cornell.edu/land/courses/ece4760/FinalProjects/s2012/cc722\\_spw54/Final/CMR\\_Robot\\_Arm.htm](http://people.ece.cornell.edu/land/courses/ece4760/FinalProjects/s2012/cc722_spw54/Final/CMR_Robot_Arm.htm)  
<http://www.superrobotica.com/Servosrc.htm>  
<http://arduino.cc/>  
[http://iearobotics.com/alberto/lib/exe/fetch.php?media=theses:memoria\\_pfc\\_brazo\\_antoniocg.pdf](http://iearobotics.com/alberto/lib/exe/fetch.php?media=theses:memoria_pfc_brazo_antoniocg.pdf)  
[http://thingiverse-production.s3.amazonaws.com/assets/78/48/67/18/63/Assembly\\_instructions.pdf](http://thingiverse-production.s3.amazonaws.com/assets/78/48/67/18/63/Assembly_instructions.pdf)

### Videos

<http://www.youtube.com/watch?v=2jQYr8tuxxM>  
<http://techcrunch.com/2012/09/05/diy-sugru-fixbot-robotic-arm/>  
<http://www.youtube.com/watch?v=eLVknHYKLhU>  
<http://www.youtube.com/watch?v=DSuKXRenlcQ&feature=related>  
<http://www.youtube.com/watch?v=d2OBCpbCEc4>  
[http://www.youtube.com/watch?v=nz\\_tgDD8FNw](http://www.youtube.com/watch?v=nz_tgDD8FNw)  
[http://www.youtube.com/watch?v=yw3ym\\_37Z0w&feature=related](http://www.youtube.com/watch?v=yw3ym_37Z0w&feature=related)  
<http://www.youtube.com/watch?v=HMQ4u9UIPSQ&feature=related>  
[http://www.youtube.com/watch?v=xJcz3\\_En\\_8o](http://www.youtube.com/watch?v=xJcz3_En_8o)  
<http://youtu.be/ThLMGLlVAQ4?t=43s>  
<http://www.youtube.com/watch?v=5Bc7gxwlUus>  
<http://www.youtube.com/watch?v=rO6vjgYT8A>  
<http://www.youtube.com/watch?v=Cd3-EnzzDQw>  
<http://www.youtube.com/watch?v=9Qo9KFS1kMo>  
<http://www.youtube.com/watch?v=fzWpMFS4VIY>  
<http://www.youtube.com/watch?v=tkDbmWAyHYw>  
[http://www.youtube.com/watch?v=W\\_gxLKSsSIE](http://www.youtube.com/watch?v=W_gxLKSsSIE)  
<http://www.youtube.com/watch?v=Wo98lDVk5dk>  
<http://www.youtube.com/watch?v=olipkRWnMH4>  
<http://www.youtube.com/watch?v=k0QjuULg2xA>  
<http://www.youtube.com/watch?v=M8f3tqY5ZC8>  
<http://www.youtube.com/watch?v=fyGalCsQ47U>  
<http://www.youtube.com/watch?v=dzHLX1eMF4A>  
<http://www.youtube.com/watch?v=FavPExqzv4>  
<http://www.youtube.com/watch?feature=endscreen&NR=1&v=zZwnyZPkylk>  
<http://www.youtube.com/watch?v=KsPqWi8W5RQ>  
<http://www.youtube.com/watch?v=BNnTyrpuFUY>  
<http://www.youtube.com/watch?v=JnvWSHTnrqo>

### Compra

[https://www.hobbyking.com/hobbyking/store/\\_37407\\_Arduino\\_Leonardo\\_R3\\_Board\\_Micro\\_USB.html](https://www.hobbyking.com/hobbyking/store/_37407_Arduino_Leonardo_R3_Board_Micro_USB.html) (Arduino Leonardo)

[http://www.hobbyking.com/hobbyking/store/\\_11855\\_hobbyking\\_928bb\\_servo\\_2\\_0kg\\_9g\\_0\\_13sec.html](http://www.hobbyking.com/hobbyking/store/_11855_hobbyking_928bb_servo_2_0kg_9g_0_13sec.html) (servo mano)

[http://www.hobbyking.com/hobbyking/store/\\_16266\\_hk15288a\\_analog\\_bb\\_mg\\_servo\\_51g\\_9kg\\_0\\_20s.html](http://www.hobbyking.com/hobbyking/store/_16266_hk15288a_analog_bb_mg_servo_51g_9kg_0_20s.html) (servo beta)

[http://www.hobbyking.com/hobbyking/store/\\_16643\\_vs\\_11\\_vigor\\_extra\\_large\\_servo\\_19sec\\_19kg\\_1\\_03g.html](http://www.hobbyking.com/hobbyking/store/_16643_vs_11_vigor_extra_large_servo_19sec_19kg_1_03g.html) (servo alfa)

[https://www.hobbyking.com/hobbyking/store/\\_27725\\_Turnigy\\_TGY\\_R5180MG\\_180\\_Degree\\_Metal\\_Gear\\_Analog\\_Servo.html](https://www.hobbyking.com/hobbyking/store/_27725_Turnigy_TGY_R5180MG_180_Degree_Metal_Gear_Analog_Servo.html) (servos omega y gamma)

<http://www.leapto3d.com/t/categorias/filamento-impresion-3d/abs-3mm> (filamentos ABS)

<http://dx.com/> (otra tienda de características y productos similares)

### Programas

<http://arduino.cc/en/Main/Software> (Software Arduino)

<http://www.microsoft.com/es-es/download/details.aspx?id=10019> (Visual Basic)

[http://www.plm.automation.siemens.com/en\\_us/academic/resources/solid-edge/index.shtml](http://www.plm.automation.siemens.com/en_us/academic/resources/solid-edge/index.shtml) (Solid Edge)

### Sitios de interés

<http://www.thingiverse.com/>  
<https://tinkercad.com/>  
<http://aivrobotics.blogspot.com/>  
<http://es-aivrobotics.blogspot.com/>

## Anexos

Planos de las piezas:

## Nota del autor

Gracias a este proyecto he tenido la oportunidad de aprender y trabajar sobre algo que me apasiona y espero que este brazo robótico pueda ser de utilidad a alguna persona. Me gustaría que este proyecto pudiera también servir de ejemplo a otras personas para que creen algo, por pequeño que sea, y lo aporten a la sociedad sin pensar en beneficios económicos. Internet nos ha dado mucho, es hora de devolver algo.

*Adrián de la Iglesia Valls*

