

Aplicación móvil para el control de un brazo robot

Mobile application to control a robot arm

Jesús Medina Cervantes

Universidad Veracruzana

jemedina@uv.mx

Nereyda Castro Gutiérrez

Universidad Veracruzana

nercastro@uv.mx

Edgar Mejía Sánchez

Universidad Veracruzana

edmejia@uv.mx

Rubén Villafuerte Díaz

Universidad Veracruzana

rvillafuerte@uv.mx

Resumen

Se presenta el desarrollo de una aplicación móvil destinada al control de un brazo robot. La aplicación se elaboró mediante la herramienta de programación basada en bloques MIT App Inventor (Friedman & Abelson, 2009), la cual permite desarrollar aplicaciones para dispositivos móviles que funcionan bajo el sistema operativo Android (Google, 2007). Para emplear la aplicación, se requiere hacer uso del módulo bluetooth con que cuenta el dispositivo móvil para poder enviar las órdenes de forma inalámbrica hacia la tarjeta de control del brazo robot. El brazo robot que es controlado por la aplicación móvil fue diseñado específicamente para este proyecto; sin embargo, se puede adaptar con facilidad el control a otros brazos robot, el único requisito es que estos cuenten con la recepción de datos por puerto serial y un módulo bluetooth para recibir la información. La tarjeta de control empleada en este proyecto para controlar las articulaciones del brazo robot es Arduino Uno (Arduino, 2005). La interfaz gráfica de la aplicación es intuitiva, de fácil manejo y cuenta con tres modos de operación, que son: manual,

programación y automático. Tanto la aplicación móvil como el brazo robot tienen la finalidad de apoyar el aprendizaje de estudiantes que cursan la carrera de Ingeniería Mecatrónica, en temas de robótica y la creación de interfaces de control con diferentes tecnologías.

Palabras clave / key words: App Inventor, Android, Arduino, bluetooth, brazo robot / App Inventor, Android, Arduino, bluetooth, robot arm.

Introducción

El desarrollo de dispositivos móviles se ha incrementado de manera exponencial en los últimos años y al mismo ritmo se ha incrementado el desarrollo de aplicaciones móviles con fines muy diversos, como entretenimiento, compras, finanzas, estilo de vida, juegos, fotografía y educación, por mencionar algunas categorías. Así mismo, los entornos de desarrollo para la creación de aplicaciones móviles van en aumento, por ejemplo Basic4Android, Mono, MIT App Inventor, LiveCode, HTML5, etc. Cada entorno de desarrollo emplea lenguajes de programación diferentes, por lo que los desarrolladores cuentan con una amplia gama de posibilidades. En particular, MIT App Inventor es una herramienta totalmente visual, pensada en aquellos usuarios que cuentan con poca o nula habilidad para programación.

Lo anterior abre la posibilidad de que estudiantes de la carrera de Ingeniería Mecatrónica puedan desarrollar aplicaciones móviles para el control de sistemas mecatrónicos con relativa facilidad, como robots móviles, brazos robots, casas inteligentes, etc.

En el presente artículo se aborda la creación de una aplicación para el control de un brazo robot, destinada para dispositivos móviles que operan bajo el sistema operativo Android, utilizando el entorno de desarrollo de App Inventor.

DISEÑO DEL BRAZO ROBOT

El diseño del brazo robot se realizó utilizando el software de diseño paramétrico SolidWorks (Dassault Systèmes, 2015). Mediante este software, en el modo “Pieza” se crearon todos los elementos del brazo

robot como objetos sólidos en tres dimensiones. Cada objeto se guarda en un archivo independiente con extensión *.sldprt. Cada uno de los elementos se creó dibujando su perfil bidimensional, indicando todas sus dimensiones mediante cotas. Después se aplicó la operación “extrusión” para obtener el objeto tridimensional deseado. Para realizar cortes o perforaciones a los objetos sólidos se empleó la operación “extruir corte”. En la figura 1 se muestra un ejemplo de este proceso.

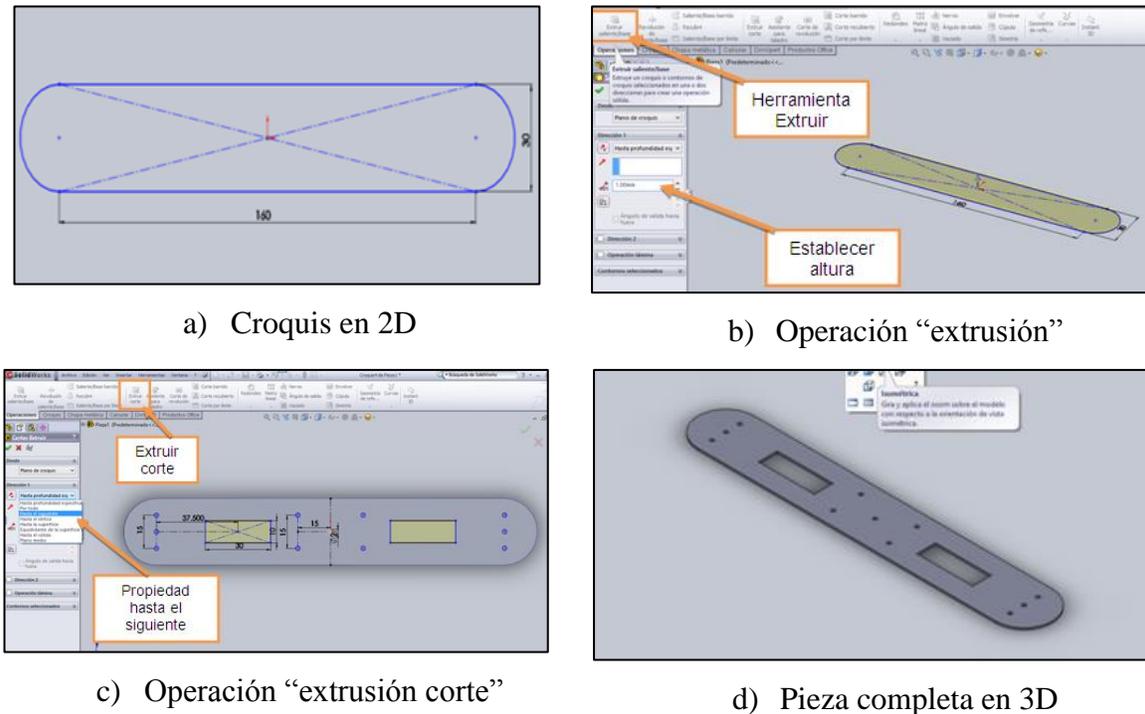
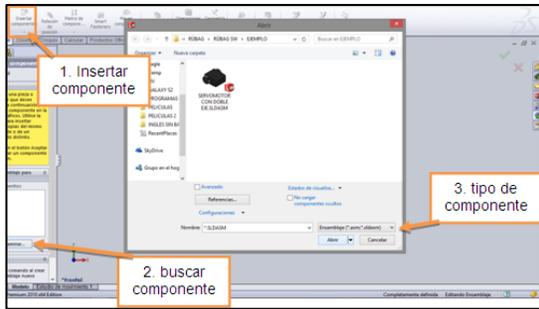


Figura 1. Proceso de modelado de un eslabón del brazo robot mediante SolidWorks.

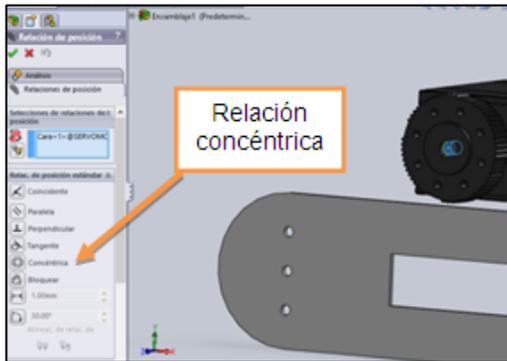
Para obtener el modelo completo del brazo robot fue necesario realizar 23 piezas diferentes, donde algunas de ellas se repiten para formar al brazo robot, dando así un total de 51 elementos. El brazo robot se desarrolla en SolidWorks trabajando en el modo “Ensamblaje”, en el cuál se emplea un solo archivo con extensión *.sldasm. El proceso de construcción del brazo robot requiere que se vayan insertando cada una de las piezas y, para que éste tome forma, se ensamblan unas con otras mediante la herramienta “Relación de posición”. En la figura 2 se ejemplifica este proceso, en la cual se observa al modelo del brazo robot completo.



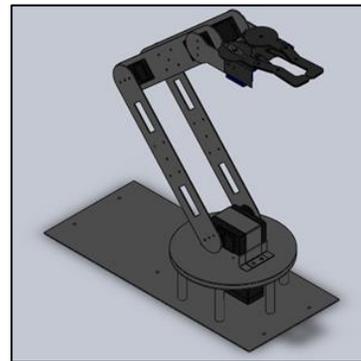
a) Se inserta una pieza



b) Herramienta "Relación de posición"



c) Se aplica relación concéntrica



d) Brazo robot diseñado en SolidWorks

Figura 2. Proceso de ensamblaje de las piezas que conforman al brazo robot.

La creación de modelos tridimensionales en SolidWorks presenta varias ventajas, de entre las cuales se pueden mencionar que al ser un software de diseño paramétrico se tiene la posibilidad de modificar las partes de manera automática al cambiar sus dimensiones, obtener los planos bidimensionales de cada una de las partes, realizar estudios de movimiento para verificar el modelo y observar posibles colisiones entre elementos, también se pueden realizar análisis de esfuerzos y deformaciones, hacer renderizados que permiten visualizar los modelos de forma realista, crear videos de ensamble y desensamble de piezas, etc. En este proyecto los planos bidimensionales facilitaron la construcción del brazo robot.

SELECCIÓN DE LOS SERVOMOTORES

Los servomotores se seleccionaron en función del torque requerido por cada una de las articulaciones del brazo robot. Para determinar los torques se empleó el modelo dinámico para robots manipuladores,

el cual ha sido descrito ampliamente por diversos autores como Spong (Spong & Vidyasagar, 1989), Reyes (Reyes, Robótica: Control de robots manipuladores, 2011) o Craig (Craig, 2006), por mencionar sólo algunos. El modelo dinámico hace uso de las ecuaciones de movimiento de Euler-Lagrange para calcular el torque.

$$\tau = M(q) \ddot{q} + \dot{M}(q) \dot{q} - \frac{\partial}{\partial q} \left[\frac{1}{2} \dot{q}^T M(q) \dot{q} \right] + \frac{\partial U(q)}{\partial q} + f_f(\dot{q}, f_e)$$

Donde:

τ es el torque requerido por la articulación.

q es el vector de posiciones articulares.

\dot{q} es el vector de velocidades articulares.

\ddot{q} es el vector de aceleraciones articulares.

$M(q)$ es la matriz de inercia.

$\dot{M}(q) \dot{q} - \frac{\partial}{\partial q} \left[\frac{1}{2} \dot{q}^T M(q) \dot{q} \right]$ es la matriz de fuerzas centrípetas y de Coriolis.

$\frac{\partial U(q)}{\partial q}$ es el vector de pares gravitacionales.

$f_f(\dot{q}, f_e)$ es el vector de pares de fricción de cada articulación.

Para conocer el torque máximo que requiere cada articulación se considera el caso extremo en el que el brazo robot se encuentra completamente extendido en posición horizontal, moviéndose a la velocidad máxima de operación. Para simplificar el cálculo, se empleó el modelo dinámico del péndulo robot, es decir un robot con un solo eslabón, ver figura 3.

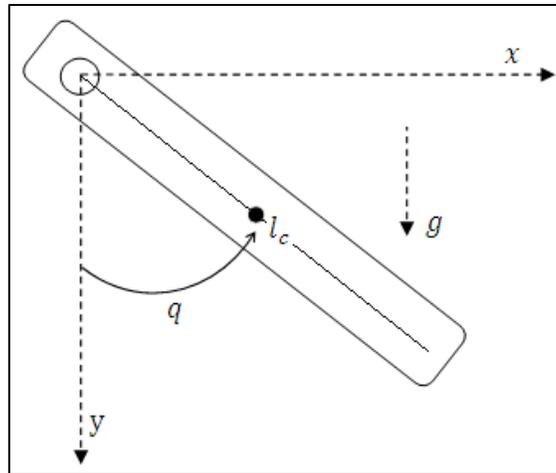


Figura 3. Péndulo robot.

De esta manera, por ejemplo, para la articulación del hombro se considera como un único eslabón o péndulo al conjunto de eslabones: brazo, antebrazo y pinza. Se requiere que el brazo robot pueda soportar una carga máxima de 50 *gramos* en la pinza. En la figura 4 se muestra un esquema del péndulo robot para la articulación del hombro, la cual será considerada con los siguientes parámetros: $m = 0.352kg$ y $l_c = 0.218 m$.

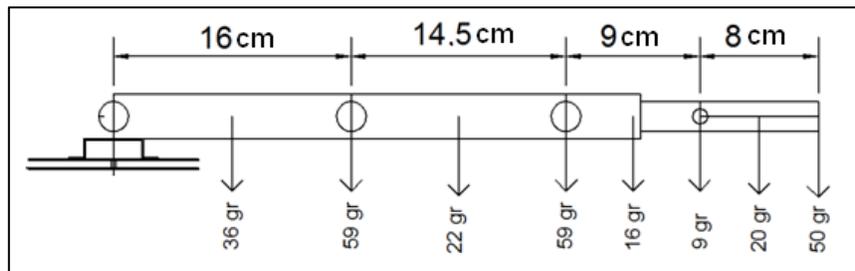


Figura 4. Péndulo robot para la articulación del hombro.

Para el caso de la articulación del codo, se considera como un único eslabón al conjunto de eslabones: antebrazo y pinza, ver figura 5. Los parámetros considerados para esta articulación son: $m = 0.176kg$ y $l_c = 0.207 m$.

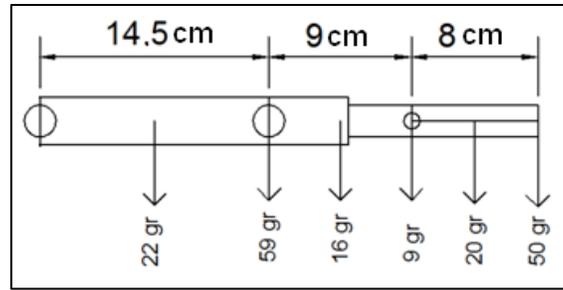


Figura 5. Péndulo robot para la articulación del codo.

Por último, para la articulación de la muñeca su único eslabón es la pinza, ver figura 6. Los parámetros considerados para esta articulación son: $m = 0.095kg$ y $l_c = 0.132 m$.

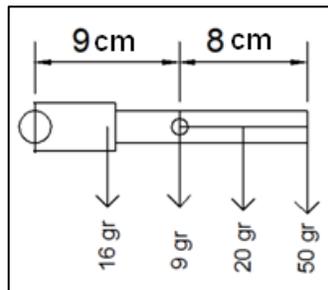


Figura 6. Péndulo robot para la articulación de la muñeca.

La ecuación empleada para el cálculo es la siguiente (Reyes, Robótica: Control de robots manipuladores, 2011):

$$\tau = [m l_c^2 + I] \ddot{q} + m g l_c \text{sen}(q) + b \dot{q} + f_c \text{signo}(\dot{q}) + f_e [1 - |\text{signo}(\dot{q})|]$$

Para los cálculos de los torques correspondientes se emplearon los siguientes datos: coeficientes de fricción para engranajes de cobre $f_c = 0.36$ y $f_e = 0.53$ (Sears, Zemansky, & Young, 1999), coeficientes de fricción para engranajes de nylon $f_c = 0.21$ y $f_e = 0.23$ (MIDSA, 2015), coeficiente de fricción viscosa $b = 0.17$ (Reyes, Robótica: Control de robots manipuladores, 2011), $q = \frac{\pi}{2}$, $\dot{q} = 1.04 \frac{rad}{s}$ y $\ddot{q} = 2.61 \frac{rad}{s^2}$.

Para determinar el torque requerido por la articulación de la base del brazo robot se empleó el modelo dinámico de una centrífuga, cuya ecuación es la siguiente (Reyes, Robótica: Control de robots manipuladores, 2011):

$$\tau = [m l_c^2 \text{seno}(\varphi) + I] \ddot{q} + b \dot{q} + f_c \text{signo}(\dot{q}) + f_e [1 - |\text{signo}(\dot{q})|]$$

En la tabla 1 se muestran los valores calculados de los torques requeridos bajo las condiciones de operación mencionadas.

Tabla 1. Torque requerido por las articulaciones del brazo robot.

Articulación	Torque requerido (kg · cm)
Hombro	14.26 kg · cm
Codo	9.41 kg · cm
Muñeca	6.73 kg · cm
Base	2.89 kg · cm

CINEMÁTICA DIRECTA E INVERSA DEL BRAZO ROBOT

En las figuras 7 y 8 se muestran los parámetros empleados por el algoritmo de Denavit-Hartenberg para obtener el modelo cinemático directo, así como el método gráfico para obtener el modelo cinemático inverso del brazo robot. La descripción completa y los resultados de la cinemática directa e inversa para un brazo robot antropomórfico de cuatro grados de libertad ya han sido presentados con anterioridad (Medina, Villafuerte, & Mejía, Simulador 3D para brazo robot de 4 grados de libertad, 2014), y para este brazo robot en particular (Reyes J. M., 2015).

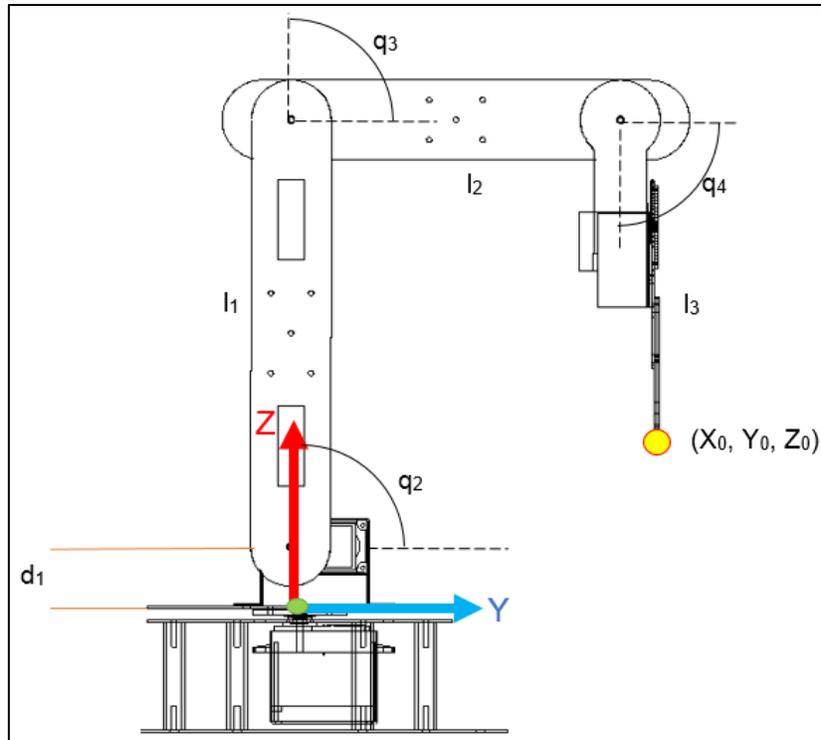


Figura 7. Parámetros del brazo robot para la cinemática directa (Reyes J. M., 2015).

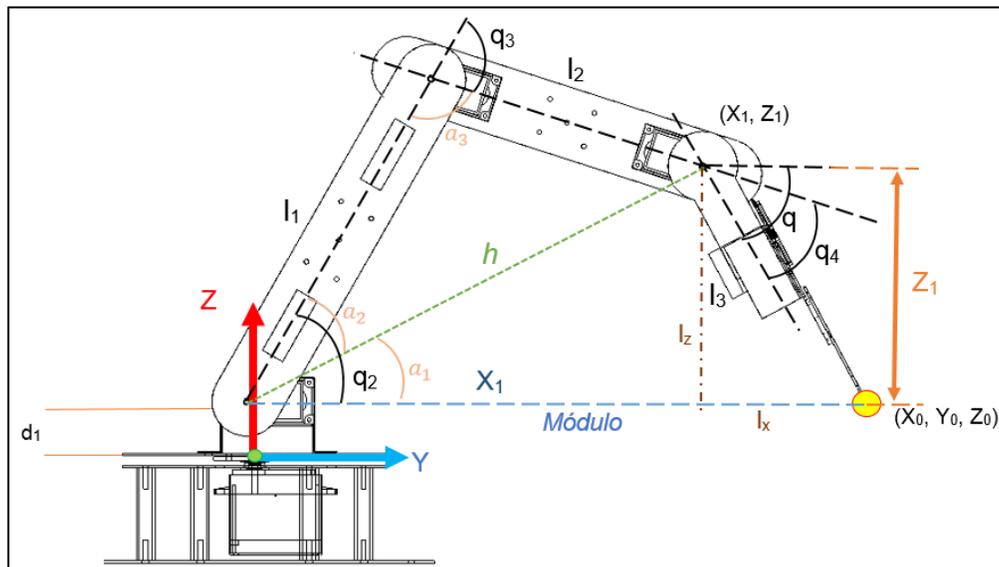


Figura 8. Parámetros del brazo robot para la cinemática inversa (Reyes J. M., 2015).

CONSTRUCCIÓN DEL BRAZO ROBOT

Los elementos del brazo robot fueron construidos empleando placa de aluminio de 1mm de espesor. A continuación se describe de manera breve la metodología empleada para la construcción del brazo robot.

Se tomaron como base los planos en 2D generados en SolidWorks y se imprimieron en una hoja de papel couché todos los elementos en escala 1:1. En la figura 9 se muestra un ejemplo de ello.

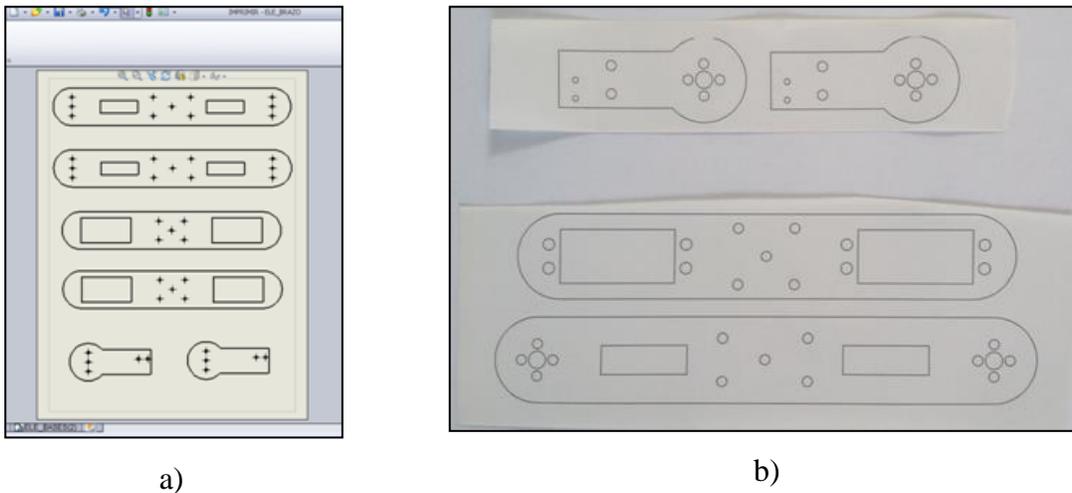


Figura 9. Elementos del brazo robot, a) planos en 2D, b) Impresión de elementos (Reyes J. M., 2015).

Después se colocó la cara impresa del papel couché sobre la placa de aluminio y esta fue planchada durante algunos minutos, con la finalidad de transferir la tinta impresa sobre la placa de aluminio para que esta sirva de guía para la realización de los cortes. Para realizar el corte del material y su acabado se utilizó una segueta y un MotoTool con su kit de herramientas. En la figura 10 se muestra el proceso de corte realizado.



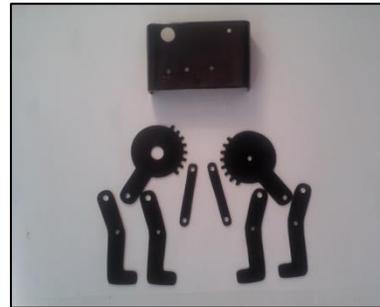
a)



b)



c)



d)

Figura 10. Corte y acabado de los elementos del brazo robot, a) corte burdo con segueta, b) corte y limado con MotoTool, c) eslabones, d) elementos de la pinza (Reyes J. M., 2015).

Teniendo todos los elementos cortados, se procedió a realizar el ensamblaje del brazo robot. En la figura 11 se muestran algunas etapas de este proceso.



a)



b)



c)



d)



e)



f)

Figura 11. Ensamblaje del brazo robot, a) base, b) antebrazo, c) brazo y antebrazo, d) pinza, e) antebrazo y pinza, f) brazo robot completo (Reyes J. M., 2015).

CONTROL DEL BRAZO ROBOT

Para realizar el control del brazo robot, en este proyecto se decidió emplear la plataforma de desarrollo de hardware libre Arduino Uno (Arduino, 2005), la cual ha tenido un incremento muy importante en su popularidad en años recientes, principalmente porque existe una gran comunidad desarrollando librerías y hardware que facilitan enormemente el desarrollo de un sistema mecatrónico o embebido. En la figura 12 se muestra esta plataforma.

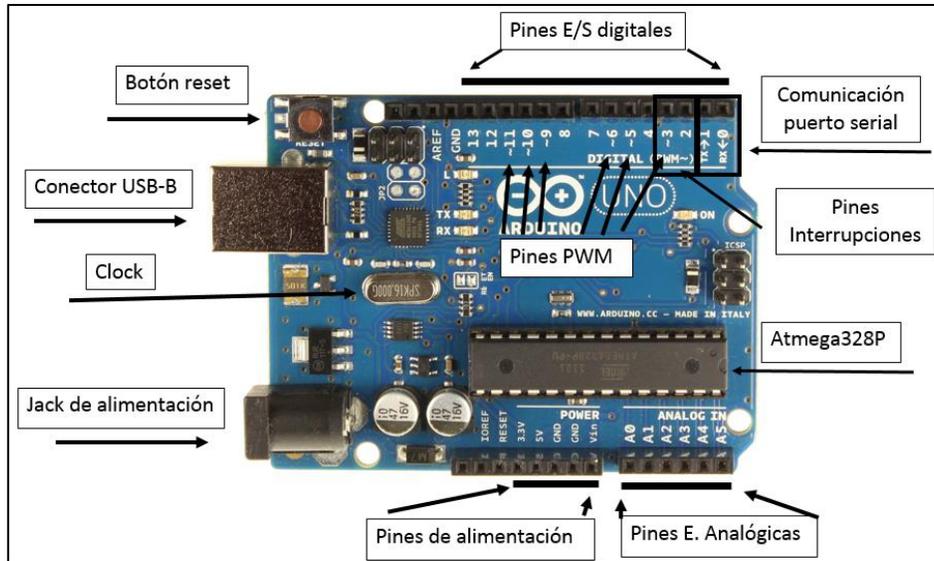


Figura 12. Plataforma de desarrollo Arduino Uno (Arduino, 2005).

Como puede verse en la figura 12, la placa Arduino Uno cuenta con pines hembra en los cuales se conectan los alambres necesarios para interactuar con el sistema a controlar, tanto entradas como salidas. La placa Arduino enviará señales de control a los servomotores y también tendrá comunicación serial a través de un módulo bluetooth HC-06, mediante la cual recibirá las órdenes de la aplicación Android. En la figura 13 se muestra un diagrama de flujo de la operación que tendrá la placa de control.

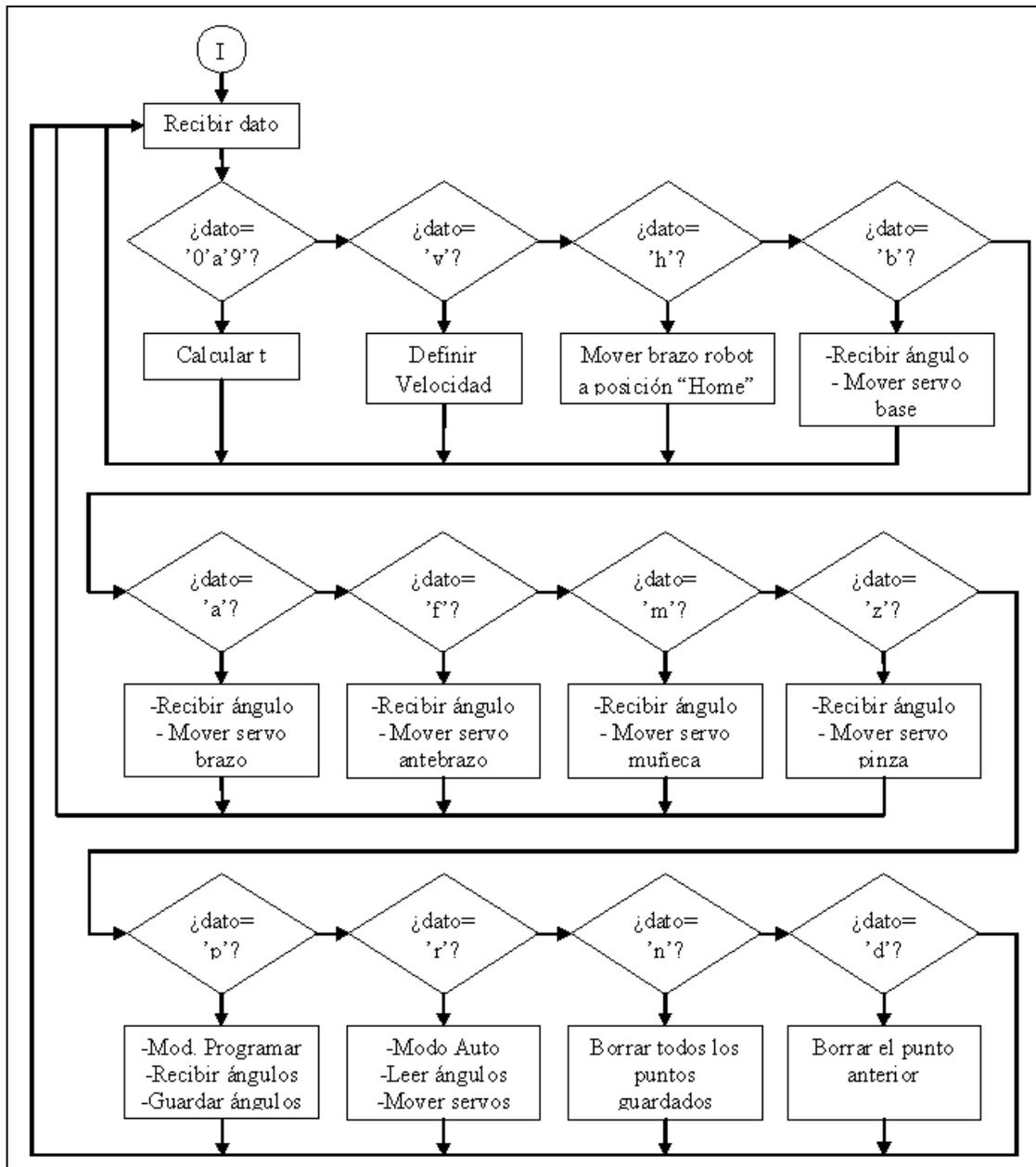
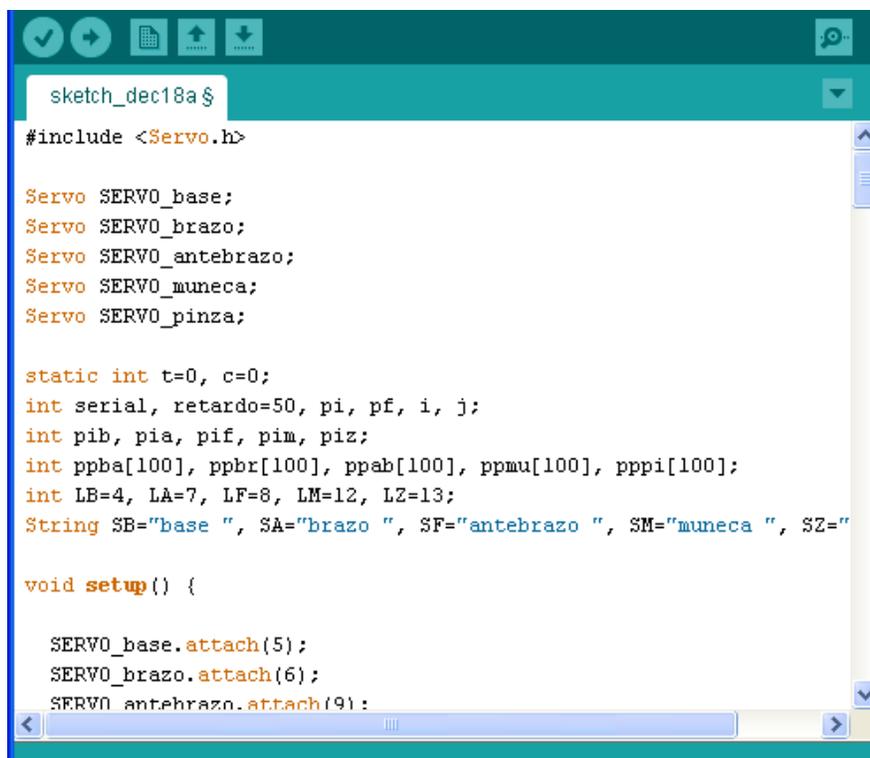


Figura 13. Diagrama de flujo del funcionamiento de la placa de control.

Arduino provee su propia interfaz de desarrollo, en la cual se realiza la programación de la placa de control. La versión utilizada en este proyecto fue la 1.0.5-r2. En la figura 14 se muestra como ejemplo la primera parte del sketch realizado para controlar las articulaciones del brazo robot.



```
sketch_dec18a$  
#include <Servo.h>  
  
Servo SERVO_base;  
Servo SERVO_brazo;  
Servo SERVO_antebrazo;  
Servo SERVO_muneca;  
Servo SERVO_pinza;  
  
static int t=0, c=0;  
int serial, retardo=50, pi, pf, i, j;  
int pib, pia, pif, pim, piz;  
int ppba[100], ppbr[100], ppab[100], ppmu[100], pppi[100];  
int LB=4, LA=7, LF=8, LM=12, LZ=13;  
String SB="base ", SA="brazo ", SF="antebrazo ", SM="muneca ", SZ=""  
  
void setup() {  
  
  SERVO_base.attach(5);  
  SERVO_brazo.attach(6);  
  SERVO_antebrazo.attach(9);
```

Figura 14. Sketch de la placa de control del brazo robot.

PLACA TIPO SHIELD

Con la finalidad de que las conexiones de salida para los servomotores y la comunicación bluetooth queden fijas, se desarrolló una placa propia tipo Shield que se ensamblará sobre la placa Arduino Uno, tal como cualquier otra placa Shield comercial.

Para desarrollar la placa Shield se empleó el software libre de diseño electrónico EAGLE (CadSoft, 2015), el cual posee una librería dedicada a Arduino. Los componentes empleados para la placa Shield se muestran en la tabla 2.

Tabla 2. Componentes electrónicos de la placa Shield.

Elemento	Nombre o Valor	Cantidad
Resistencia	330 Ω	5
Led	Verde	5
Pin macho	90°	19
Pin macho	Recto	16
Bornera	2 terminales	1

En la figura 15 se muestra el diagrama esquemático de la placa Shield realizada en Eagle.

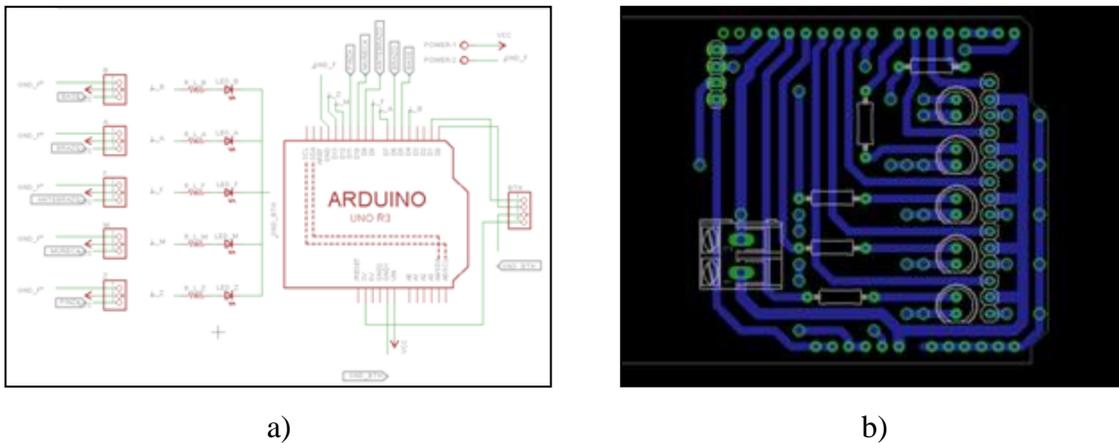
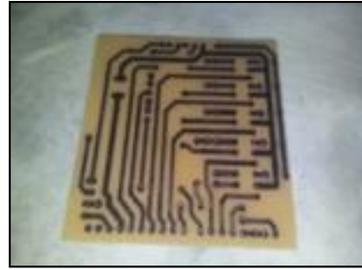


Figura 15. Diseño de la placa Shield en EAGE, a) modo esquemático, b) modo placa (Reyes J. M., 2015).

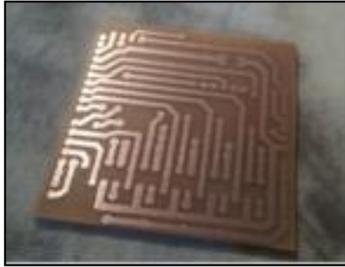
Para construir la placa tipo Shield se realizaron los siguientes pasos. Primero se imprimió el circuito electrónico en una hoja de papel couché, en seguida se colocó el lado impreso sobre una placa fenólica y se planchó este durante algunos minutos, con lo cual se transfirió el circuito impreso sobre la placa. Después se eliminó el papel restante con agua. El siguiente paso fue la eliminación del cobre de la placa para que quedaran únicamente las pistas del circuito, esto se realizó bañando la placa en una mezcla de cloruro férrico y agua durante algunos minutos. Enseguida se procedió a limpiar la placa y lijarla para remover el tóner de las pistas para que estas quedaran visibles. Después se perforó la placa utilizando un MotoTool y brocas de 1/32” y 1/16”. Por último, se colocaron y soldaron todos los componentes para completar la placa Shield. En la figura 16 se muestra este proceso.



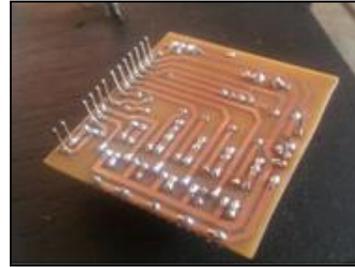
a)



b)



c)



d)



e)



f)

Figura 16. Construcción de la placa Shield, a) remoción de cobre, b) pistas de placa Shield, c) lijado de pistas, d) soldadura de componentes, e) placa Shield, f) placa Shield ensamblada sobre Arduino Uno (Reyes J. M., 2015).

APLICACIÓN MÓVIL

La aplicación móvil que fue desarrollada en este proyecto funciona bajo el sistema operativo Android, el cual es considerado actualmente como el sistema operativo para dispositivos móviles más popular del mundo. La aplicación móvil fue desarrollada mediante la herramienta de programación basada en bloques MIT App Inventor (Friedman & Abelson, 2009), la cual es una plataforma de código abierto.

El primer objetivo de la aplicación móvil es establecer comunicación inalámbrica con la placa de control del brazo robot. Para lograrlo, se requiere hacer uso del módulo bluetooth con que cuenta el dispositivo móvil. El segundo objetivo es enviar las órdenes hacia la tarjeta de control del brazo robot.

Para desarrollar la aplicación móvil se realizaron los pasos descritos a continuación. Primero se debe acceder al sitio web www.appinventor.mit.edu. En la página principal, dar clic en la parte superior derecha sobre el botón con la leyenda "Create apps!". En el sitio web aparece una nueva ventana en la que se pide aceptar los términos de licencia y crear una cuenta de google o emplear el correo actual. Enseguida, otra ventana da la bienvenida y pide que se llene un formulario para contar con información del tipo de usuario. Después otra ventana sugiere la lectura de la "Release Notes", en la cual se presiona el botón "Continue".

Así, se da clic sobre el botón "Start new project", con lo cual se abre una ventana donde se debe escribir el nombre de la aplicación móvil, ver figura 17.

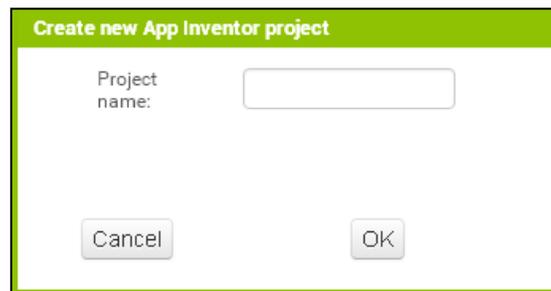


Figura 17. Nombre del proyecto de la nueva App.

Una vez creado el proyecto, se debe dar clic sobre el nombre del mismo. Enseguida aparece la página principal de desarrollo de App Inventor, ver figura 18.

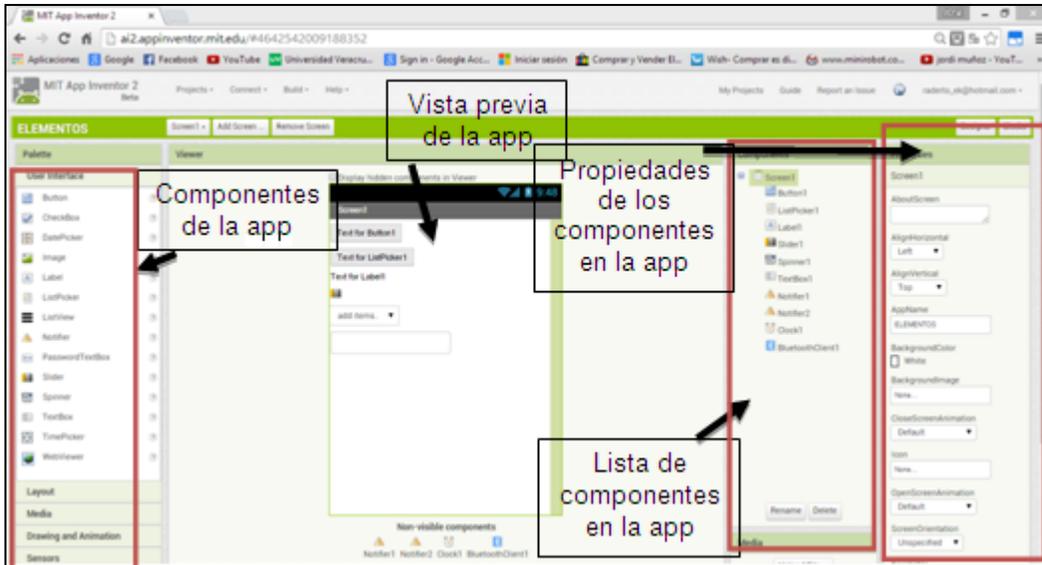


Figura 18. Ventana de diseño de Apps.

En la ventana de diseño se agregan todos los elementos necesarios para construir el layout o aspecto que se requiera que tenga la aplicación móvil. Por ejemplo, en la figura 19 se muestra el layout realizado para establecer la comunicación por bluetooth con la placa de control del brazo robot y en la figura 19, se muestran los bloques de programa correspondientes para establecer dicha comunicación.



Figura 18. Layout para establecer comunicación con el brazo robot (Reyes J. M., 2015).

```

when B_CONECTAR . Click
do
  if not BLUETOOTH_CLIENT . Enabled
  then
    call NOTIFIER_BLUETOOTH . ShowAlert
    notice "Active Bluetooth"
  else
    call LISTA_BLUETOOTH . Open

when LISTA_BLUETOOTH . BeforePicking
do
  set LISTA_BLUETOOTH . Elements to BLUETOOTH_CLIENT . AddressesAndNames

when LISTA_BLUETOOTH . AfterPicking
do
  set LISTA_BLUETOOTH . Selection to call BLUETOOTH_CLIENT . Connect
  address LISTA_BLUETOOTH . Selection
  if BLUETOOTH_CLIENT . IsConnected
  then
    set L_BLUETOOTH_ESTADO . BackgroundColor to #0000FF
    set L_BLUETOOTH_ESTADO . Text to "Bluetooth Conectado"
    set L_BLUETOOTH_ESTADO . TextColor to #FFFFFF
    set L_S2_MODALIDAD . Visible to true
    set L_S2_MODALIDAD . Text to "MENU"
    set L_S2_MODALIDAD . BackgroundColor to #FF0000
    set LAYOUT_MENU . Visible to true
    set LAYOUT_DEFINIR_ANGULO . Visible to false
    set LAYOUT_MANUAL . Visible to false
    set LAYOUT_PUNTO_A_PUNTO . Visible to false
    set B_HOME_MENU . Visible to false
    set B_CONECTAR . Visible to false
    set B_DESCONECTAR . Visible to true
    set L_BLUETOOTH_RECIBE . Visible to false
    set SP_PROGRAMAS . Elements to call DB_PROGRAMAS . GetTags
  
```

Figura 19. Bloques de programa para establecer la comunicación (Reyes J. M., 2015).

La aplicación móvil que se desarrolló en este proyecto cuenta con tres modos de operación, que son: modo manual, modo programación y modo automático. Por ejemplo, para el modo manual se tienen dos opciones de layout, una en la que los ángulos se ingresan numéricamente y otra en la que se ingresan mediante controles deslizantes. En la figura 20 se muestra el layout para ingreso numérico de los ángulos y sus bloques de programa correspondientes se muestran en la figura 21. La descripción completa de la aplicación móvil, así como de los bloques de programa, se encuentra en (Reyes J. M., 2015).



Figura 20. Layout para el modo de operación manual (Reyes J. M., 2015).



Figura 21. Bloques de programa para el modo manual (Reyes J. M., 2015).

Conclusión

En el presente artículo se mostró la metodología empleada para el desarrollo de un brazo robot didáctico de tipo antropomórfico, así como también la creación de una placa tipo Shield dedicada específicamente para ser ensamblada con Arduino Uno y la creación de la aplicación móvil para el controlar al brazo robot. App Inventor constituye una gran plataforma para el desarrollo de aplicaciones, tanto para desarrolladores principiantes como para avanzados. El uso de los bloques de programación se vuelve intuitivo conforme se practica la programación en la plataforma App Inventor. Se realizaron diversas pruebas de funcionamiento a los modos de operación y todas fueron exitosas. El desarrollo del prototipo didáctico del brazo robot y la aplicación móvil servirán como apoyo para el aprendizaje en la experiencia educativa de robótica y sistemas de comunicación para estudiantes de Ingeniería Mecatrónica de la Facultad. En otro proyecto realizado por los autores ya se abordó el desarrollo de aplicaciones móviles con otra plataforma (Medina, Santos, Mejía, & Villafuerte, 2015), por lo que este trabajo ofrece otra opción para que los estudiantes desarrollen sus propias aplicaciones y adquieran esta competencia de actualidad y de importancia creciente en el campo laboral.

Bibliografía

- Arduino. (2005). *Arduino*. Recuperado el 8 de Enero de 2015, de <https://www.arduino.cc/>
- CadSoft. (15 de Enero de 2015). *CS EAGLE*. Obtenido de <http://www.cadsoftusa.com/download-eagle/freeware/>
- Craig, J. (2006). *Robótica*. México: Pearson - Prentice Hall.
- Dassault Systèmes. (22 de Abril de 2015). *SolidWorks*. Obtenido de <http://www.solidworks.com/>
- Friedman, M., & Abelson, H. (2009). *MIT App Inventor*. Recuperado el 10 de Febrero de 2015, de <http://appinventor.mit.edu/explore/>
- Google. (2007). *Android*. Recuperado el 12 de Enero de 2015
- Medina, J., Santos, J., Mejía, E., & Villafuerte, R. (2015). Aplicaciones Android para el control de sistemas mecatrónicos. *Revista Iberoamericana de Producción Académica y Gestión Educativa*, *Publicación #2*, ISSN 2007-8412, 1-20.
- Medina, J., Villafuerte, R., & Mejía, E. (2014). Simulador 3D para brazo robot de 4 grados de libertad. *Revista Iberoamericana para la investigación y el desarrollo educativo*. *Publicación #12*. ISSN 2007-7467, 1-19.

- MIDSA. (16 de Octubre de 2015). *MIDSA*. Obtenido de <http://www.midsa.com.mx/pdf/Nylatron-GS-GSM-NSM.pdf>
- Reyes, F. (2011). *Robótica: Control de robots manipuladores*. México: Alfaomega - Marcombo.
- Reyes, J. M. (2015). *Brazo robótico controlado mediante una aplicación móvil*. Ciudad Mendoza, Ver.: Tesis de licenciatura. Universidad Veracruzana.
- Sears, F., Zemansky, M., & Young, H. (1999). *Física Universitaria, Volumen 1*. México: Pearson Education.
- Spong, M., & Vidyasagar, M. (1989). *Robot dynamics and control*. New York: John Wiley & Sons, Inc.