

Utilización de Arduino en dispositivos electrónicos Using Arduino in electronic devices

¹Palafox Rodriguez Eduardo Daniel P, ²Avila Camacho, F.J.

^{1,2}División de Ingeniería en Sistemas Computacionales, TESE - TecNM, Ecatepec de Morelos, Estado de México
, CINVESTAV-IPN, Ciudad de México, México
Teléfono (55) 92007251 E-mail: 201510708@tese.edu.mx,

ingeniería en sistemas computacionales , Tecnológico de estudios superiores de Ecatepec, Municipio Ecatepec , Estado de México, México.

Resumen

En el actual proyecto se analizara acerca del Arduino y sus diferentes variantes, en el cual nos proporciona sus usos como microcontroladores y herramientas que sirven a los desarrolladores para la creación de la robótica. ¿Alguna vez han escuchado de este hardware libre? Para que se utiliza, en donde se utiliza, que es, cuáles son sus aplicaciones, que se necesita, cuáles son sus requerimientos de software, hardware, todos estos puntos son los que vamos a analizar para que podamos conocer más acerca de esta plataforma que nos brinda la puerta abierta a esta tecnología que es parte de nuestro futuro. Podemos ver que la mayoría de los robots con grados de libertad se aplican en la industria que permiten que las tareas puedan ser automatizadas y de las cuales la mayoría son realizadas con el menor tiempo posible, que puedan ser eficaces y con mejor calidad. Ya que la demanda de un producto tenga que estar listo con el menor tiempo posible para satisfacer estas necesidades de hoy en día.

Palabras Clave— Fuente abierta, robótica, herramientas, bibliotecas, hardware, Sistema operativo, software.

Abstract

In the current project, the Arduino 1 and its different variants will be analyzed, in which it provides us with their uses as microcontrollers and tools that serve developers for the creation of robotics. Have you ever heard of this free hardware? What is it used for, where is it used, what is it, what are its applications, what is needed, what are its software and hardware requirements, hardware, all these points are what we are going to analyze so that we can learn more about this platform that gives us the open door to this technology that is part of our future. We can see that most of the robots with degrees of freedom are applied in the industry that allow tasks to be automated and most of which are performed in the shortest possible time. that can be effective and with better quality. Since the demand for a product has to be ready in the shortest possible time to meet these needs today.

Keywords — Open source, robotics, tools, libraries, hardware, operating system, software.

*Autor para la correspondencia: josejuan@dominio1.edu.mx

Correo electrónico: edudan03@gmail.com (Palafox Rodriguez Eduardo Daniel), 201510708@tese.edu.mx

I. Introducción

Arduino es una plataforma electrónica de código abierto basada en hardware y software de fácil manejo que se utiliza para la construcción de diferentes proyectos electrónicos del mismo, está formado por una tarjeta o placa física de circuito programable (normalmente denominada microcontrolador) y un software, o IDE (Integrated Development Environment) que se instala en tu ordenador, y que se utiliza para picar y cargar código del ordenador a la tarjeta física.

Las placas de Arduino se caracterizan por leer entradas – la luz de un sensor, pulsar un botón, o un mensaje de texto enviado a una Red Social – para convertirla en una salida – activando un motor, encendiendo un LED, publicando algo on-line.

II. Antecedentes

En el desarrollo real el Arduino fue inventado en 2005 por Massimo Banzi, entonces estudiante de la facultad del IVRAE, quien originalmente pensó en crear un Arduino de la misma facultad para las necesidades de enseñanza de los estudiantes de computación y electrónica, porque en ese momento era difícil obtener una placa de microcontrolador. muy costoso y no proporcionó el apoyo adecuado; sin embargo, nunca esperó que años después esta herramienta se convirtiera en líder mundial en tecnología DIY (Do It Yourself). El proyecto se creó originalmente no solo para ahorrar dinero mediante la creación de proyectos escolares en el instituto, sino que Banzi también tenía la intención de ayudar a su escuela a evitar la bancarrota utilizando las ganancias de la venta de sus placas en el campus.

Los primeros prototipos de Arduino se hicieron en el Instituto IVRAE. Originalmente era una placa simple con un microcontrolador simple y resistencias de voltaje conectadas, y solo se podían agregar sensores simples como LED u otras resistencias, y aún no admitía la programación en ningún lenguaje para manipularlo. Años más tarde, Hernando Barragán, estudiante de posgrado de la Universidad de Columbia, se unió al equipo de Arduino y, tras conocer el proyecto, ayudó a desarrollar el entorno de programación del procesador de la placa: wire, en otra institución del mismo instituto que Banzi. miembro David Mellis, quien más tarde mejoró la interfaz del software.

Arduino con Scratch -S4AScratch for Arduino (S4A) es una modificación de Scratch que facilita la programación de la plataforma de hardware libre Arduino. Proporciona nuevos bloques para procesar sensores y actuadores conectados a placas Arduino. Proporciona una interfaz de trabajo muy simple e intuitiva adecuada para varios proyectos (desde los más simples hasta los juegos o animaciones más complejos).



Fig. 1 Logotipo de Arduino S4

Qué ofrece Arduino1:

- 1- Para desarrollar elementos autónomos, conectarse a otros dispositivos o interactuar con otros programas para interactuar con hardware y software.
- 2- Funciona como un microcontrolador, debido a que el microcontrolador Arduino tiene una llamada interfaz de entrada, que es una conexión a través de la cual podemos conectar varios tipos de periféricos a la placa de circuito impreso. La información de estos periféricos conectados se transferirá al microcontrolador, que se encargará de procesar los datos que llegan a través de ellos.
- 3- Una de las ventajas que aporta es que el tipo de periférico que puedes utilizar para enviar datos al microcontrolador depende mucho del uso que le vayas a dar. Estos pueden ser cámaras para recuperar imágenes, teclados para ingresar datos o varios tipos de sensores
- 4- También cuenta con una interfaz de salida que se encarga de enviar la información procesada en el Arduino a otros dispositivos periféricos. Estos periféricos pueden ser monitores o altavoces para reproducir los datos procesados, pero también pueden ser otras placas o controladores.
- 5- También tienen otro tipo de componentes llamados escudos (escudos) o mochilas. Es una tarjeta que se conecta a la placa base para agregar una variedad infinita de funciones.
- 6- Otras funcionalidades que ofrece son la creación de dispositivos GPS, relojes en tiempo real, conexiones de radio, pantallas LCD táctiles, placas de desarrollo y elementos de diversa índole.
- 7- Arduino IDE es compatible con varios sistemas operativos como Windows, Linux o Mac. Multiplataforma, libertad, una gran comunidad, facilidad de uso son algunas de las características en las que se basa el éxito de Arduino, un claro ejemplo del uso del sistema operativo es el sistema operativo ROS, que cuenta con útiles herramientas de visualización de proyectos. o robots. Incluyen Rviz y Gazebo
- 8- Estas funciones se pueden dividir en diferentes tipos:
 - Funciones de entradas/salidas digitales (I/O: Input/Output).
 - Funcionalidad de entrada/salida analógica.
 - Funciones de entrada/salida mejoradas.
 - Funcionalidad de entrada/salida de Arduino Due.
 - Funciones temporales.
 - Funciones matemáticas.
 - Funciones trigonométricas

II.

III. COMANDOS DE ARDUINO

pinMode El código `pinMode(num_pin, comportamiento)`; permite configurar si será usado el pin como entrada o salida. En lugar de `num_pin`, se coloca el número del pin que se desea configurar. La palabra `comportamiento` debe de sustituirse por `INPUT` para usarlo como entrada. Se debe colocar `OUTPUT` si se desea que funcione como salida.

digitalRead: En segundo lugar, se utiliza la instrucción `digitalRead(num_pin)`; para leer un pin. Donde dice `num_pin` se debe indicar el número del pin que se desea leer.

digitalWrite: Finalmente, para activar o desactivar una salida se utiliza la instrucción `digitalWrite` que significa `num_pin,valor` término `num_pin` se sustituye por el número del pin a utilizar. También, donde dice `valor` se coloca `HIGH` (alto) si se desea generar un 1. En el caso contrario, si se desea generar un 0, se escribe la palabra `LOW` (bajo).

comandos para pines analógicos

analogRead Para leer un pin analógico se utiliza el comando `analogRead(pinAnalog)`; El término `pinAnalog` se debe sustituir por el pin analógico que se va a utilizar (A0–A5).

analogWrite Para la salida “analógica” o PWM se utiliza `analogWrite (pinSalida,valorSalida)`; Ahora, `pinSalida`, indica el pin donde se va a generar la señal PWM. El valor del ciclo de trabajo de la señal PWM (0-255), se indica dónde está la palabra `valorSalida` de la señal PWM (0-255) expresado por valor de palabra `Salida`.

IV. APLICACIONES

Algunos ejemplos de aplicaciones de Arduino son

- Monitoreo en tiempo real con Arduino.
- Acceso al control remoto del sistema.
- Puede darte notificaciones rápidas.
- Es muy eficiente energéticamente.
- Utilizar la automatización de procesos.
- Automatización de reportes/tableros.
- Mantenimiento predictivo.
- Potencial redox.



Ilustración 1 Tipos de Arduino

Hay muchos entornos de aplicación para Arduino: automatización industrial, domótica, herramientas de creación de prototipos, plataformas de aprendizaje para aprender electrónica, técnicas artísticas, eficiencia energética, monitoreo, recopilación de datos, bricolaje, aprendizaje de habilidades técnicas y programación en instituciones educativas de nivel medio y superior, así como de formación profesional y universidades, Arduino actúa como una poderosa herramienta para la formación y el aprendizaje de habilidades..



Ilustración 2 prototipo de robot seguidor

V. Tipos de placa de Arduino 3

Es probablemente la tabla más usada y más resistente. Una excelente opción para iniciar la programación de microcontroladores. Estas son sus características técnicas



Ilustración 3 Arduino 1

Tipos de Arduino UNO

- Microcontrolador: ATmega328P.
- Velocidad de reloj: 16 MHz.
- Voltaje de trabajo: 5V.
- Voltaje de entrada: 7,5 a 12 voltios.
- Pinout: 14 pines digitales (6 PWM) y 6 pines analógicos.
- 1 puerto serie por hardware.
- Memoria: 32 KB Flash (0,5 para bootloader), 2KB RAM y 1KB Eeprom.

Conclusiones

Existen plataformas de hardware libres como Arduino, una placa con un controlador y un entorno de desarrollo, su sencillez y bajo costo nos permiten crear más diseños y

muchos tipos de aplicaciones. usamos Arduino porque nos permite hacer muchas tareas, ahorra componentes electrónicos, pero usa un lenguaje de programación que lo hace muy fácil de usar. cuando nos asignaron un experimento con electrónica y se nos ocurrió el tema de Arduino, decidimos hacer algo que combinara la electrónica que vimos en el salón de clases con algo nuevo y único. menciona al revisor en twitter. hemos cubierto cómo hacer que el dispositivo se conecte a internet, busque tweets y, finalmente, cuando su nombre de usuario aparece en un tweet, se notifica al usuario, lo que significa que el usuario recibe una mención. la primera idea es escribir un programa de búsqueda de palabras que busque constantemente nombres de usuario. luego necesitamos evidencia o una indicación de que se ha encontrado un tweet, como un led encendido o música. el equipo concluyó que la forma más sencilla de combinar algunos lenguajes de programación.

REFERENCIAS

1. Evans, Brian W. (2007) Arduino Programación NOTebok
2. Santo Oncero David (2007) Free>, All Linux, Madrid: Studio Press. Sábado 21-12
3. Página de inicio de Arduino: <http://www.arduino.cc/es/>
4. Sistema Microcontrolador y Programación Juan Manuel Orduña Huertas, Vicente Arnau Llombar. Universidad de Valencia 1996.
5. Microcontroladores: Fundamentos y Aplicaciones de PIC. Ramón Palas Arena. Editorial 3T.
6. Microcontrolador PIC: un sistema integrado de autoaprendizaje. MARCOMBO, edición técnica 2007, MARCOMBO S.A. Enrique Mandado Pérez, Luis Menéndez Fuertes
7. CIM, Informática y Automatización de la Fabricación Andrés García Higuera, Fernando J. Castillo García. Publicación de la Universidad de Castilla-La Mancha.
8. Microcontrolador PIC: sistema integret selvlærende. MARCOMBO, Edición Técnica 2007, MARCOMBO S.A. Enrique Mandado Pérez, Luis Menéndez Fuertes.
9. Redes de área personal inalámbricas de baja velocidad: uso de IEEE 802.15.4 para habilitar sensores inalámbricos José A. Gutiérrez, Edgar H. Callaway, Raymond L. Barret. Red de información estándar.
10. Secuencia del motor DC. Gilbert Henry Arpa. Limusa, Prensa Noriega.