

“INTRODUCCIÓN A LA PROGRAMACIÓN EN ARDUINO UTILIZANDO EL ENTORNO DE DESARROLLO INTEGRADO (IDE).”

Informe N°2

Laboratorio de Sistemas Embebidos

Melanny Dávila
Ingeniería en Telecomunicaciones
Facultad de Eléctrica y Electrónica
Quito, Ecuador
melanny.davila@epn.edu.ec

2nd Jonathan Álvarez
Ingeniería en Telecomunicaciones
Facultad de Eléctrica y Electrónica
Quito, Ecuador
jonathan.alvarez@epn.edu.ec

Abstract—A medida que Arduino ha ganado popularidad en ámbitos educativos y de aplicaciones para IoT, además del aumento de módulos disponibles se ha visto la necesidad de mejorar el IDE de Arduino, que cada vez ha ido implementando mayor cantidad de librerías para usos más amplios y especializados. El IDE nos permite crear, compilar, testear programas y funciones que deseemos implementar en Arduino. Cuenta con compatibilidad con múltiples versiones de Arduino y nos permitirá interactuar con la placa de una manera efectiva.

Index Terms—Arduino, IDE, consola, menú.

I. INTRODUCCIÓN

Arduino IDE es un entorno para desarrollar la programación necesaria para el funcionamiento de las placas de Arduino; este entorno ha sido empaquetado como un programa de aplicación. Consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI) [1].

Arduino cuenta con una infinidad de librerías que facilitan la creación de un proyecto, gran porcentaje del código más complejo se encuentra resuelto en librerías, las cuales pueden ser usadas con gran facilidad.

II. OBJETIVOS

- Relacionar al estudiante con los fundamentos de los Sistemas Embebidos.
- Establecer e identificar características generales de la programación en el IDE de Arduino.
- Establecer una comparación entre el IDE de Arduino y la herramienta de simulación Tinkercad [2].

III. CUESTIONARIO

A. Realizar un cuadro comparativo de las ventajas y desventajas de la programación en el IDE de Arduino y la programación de Bloques en Tinkercad.

Programación en el IDE de Arduino	
Ventajas	Desventajas
No es necesaria tener una conexión a internet para realizar el desarrollo de programas.	La escritura de código no es intuitiva si no se tiene conocimientos previos en programación
Se puede agregar o crear gran numero de funcionalidades haciendo uso de las librerías o escribiendo nuevas librerías.	Se debe tener en cuenta buenas practicas de programación para evitar códigos muy confusos
Es posible utilizar gran cantidad de términos y funciones.	Es recomendable el conocimiento previo de otros lenguajes de programación
La codificación es muy completa y permite llevar a cabo gran cantidad de proyectos complejos	Gran cantidad de funciones hace necesario para llevar a cabo proyectos grandes el uso de documentación
Existe una ventana que nos informa del estado de la placa y existe otra que nos informa de los errores cometidos	

Programación de Bloques en Tinkercad	
Ventajas	Desventajas
Los bloques de código son fáciles de utilizar	El diseño de proyectos complejos se vuelve complicado
La interfaz grafica es intuitiva	La navegación en bloques extensos se vuelve confusa
No requiere conocimientos extensos en programación	Carece de funciones complejas para aplicaciones profesionales
Permite crear programas rápidos o sencillos de manera fácil	Requiere conexión internet para poder acceder a la plataforma
Cuenta con herramientas básicas muy completas	Es necesario registrarse para poder usar el servicio
Permite exportar los bloques de como código	La interfaz del navegador se puede ver limitada si no se tiene una computadora buena
Se puede ingresar desde cualquier parte si se cuenta con conexión a internet	

B. Describir la importancia del uso de Librerías en Arduino y describir 5 aplicaciones relacionadas a su carrera donde se utilicen librerías.

La instalación, actualización y manejo librerías en Arduino es un aspecto importante a la hora de usar Arduino debido a que facilitan mucho la programación y permite la abstracción haciendo que el programa sea más sencillo de hacer y de entender [5].

Además, las librerías en Arduino son importantes ya que estas agregan funcionalidades para controlar dispositivos como sensores o pueden contener funciones que no vienen por defecto en el IDE y bajo otras circunstancias tendrían que ser escritas desde cero. En otras palabras, su importancia recae en que resumen procesos y permiten mejorar las funciones que tiene la placa [6].

Las librerías normalmente incluyen los siguientes archivos comprimidos en un archivo ZIP o dentro de un directorio. Estas siempre contienen:

- Un archivo .cpp (código de C++)
- Un archivo .h o encabezado de C, que contiene las propiedades y métodos o funciones de la librería.
- Un archivo Keywords.txt, que contiene las palabras clave que se resaltan en el IDE (opcional).
- Muy posiblemente la librería incluye un archivo readme con información adicional de lo que hace y con instrucciones de como usarla.
- Directorio denominado examples con varios sketches de ejemplo que nos ayudará a entender cómo usar la librería (opcional) [7].

A continuación se presentan 5 tipos de librerías presentes en Arduino con sus respectivas aplicaciones en la carrera de Telecomunicaciones.

- **Librería GSM:** permite que una placa Arduino realice la mayoría de las operaciones que puede realizar con un teléfono GSM: realizar y recibir llamadas de voz, enviar y recibir SMS y conectarse a Internet a través de una red GPRS [8]. Usada al momento de desarrollar redes de comunicaciones móviles.
- **Librería FFT:** es una implementación rápida de un algoritmo FFT estándar que opera solo con datos reales. Puede proporcionar hasta 256 bandejas de frecuencia a 16b de profundidad, con una tasa de actualización mínima de aproximadamente 7ms. Se puede ajustar de 16 a 256 bandejas y tiene varios métodos de salida para satisfacer diversas necesidades. Se puede configurar en salida de 16b lineal, 8b lineal, 8b logarítmica u 8b octava [9]. Esta librería es muy útil en cada procesamiento digital de señales en cualquier sistema de comunicación.
- **Librería Audio Frequency Meter:** permite que un Arduino basado en arquitecturas SAMD (es decir (es decir, Zero, MKRZero o MKR1000 Board)) muestree una señal de entrada analógica y recupere su frecuencia. Se puede elegir el pin analógico que se utilizará para muestrear la señal y la frecuencia de muestreo [10].

- **Librería Ethernet:** permite que una placa Arduino se conecte a Internet con el fin de poder realizar simulaciones de red. La placa puede servir como servidor que acepta conexiones entrantes o como cliente que realiza conexiones salientes. La biblioteca admite hasta ocho (W5100 y las placas con ≤ 2 kB SRAM están limitadas a cuatro) conexiones simultáneas (entrantes, salientes o una combinación) [11].
- **Librería XBee:** permite comunicarse con XBees en modo API, con soporte para Series 1 (802.15.4) y Series 2 (ZB Pro/ZNet). Esta biblioteca incluye soporte para la mayoría de los tipos de paquetes, incluidos: TX/RX, comando AT, AT remoto, muestras de E/S y estado del módem [12].

C. Conclusiones:

Jonathan Álvarez

- Se puede ampliar el alcance de las aplicaciones desarrolladas en Arduino a través de librerías que pueden ser provistas por los fabricantes de dispositivos o pueden ser creadas por el usuario de necesitarlo.
- Es posible importar el código de la plataforma Tinkercad a el IDE de Arduino, y viceversa pero debemos tener en cuenta la compatibilidad en este caso.
- Ambas plataformas tienen sus ventajas y desventajas, por lo tanto debemos usarlas dependiendo de la aplicación que se quiera desarrollar, teniendo en cuenta que para aplicaciones complejas es mejor utilizar el IDE de Arduino

Melanny Dávila

- El uso de librerías en el desarrollo de un proyecto, mejora e incrementa la cantidad de usos que se le puede dar a cualquier placa Arduino.
- La instalación de una librería es muy útil debido a que dentro del IDE de Arduino se puede encontrar documentación y ejemplos de código de su uso.
- El uso del IDE de Arduino es recomendado cuando se realiza una aplicación extensa de las funciones de la placa Arduino ya que este entorno es capaz de almacenar bastante información.

D. Recomendaciones:

Jonathan Álvarez

- Revisar que en las librerías instaladas se cuente con el modelo específico del dispositivo que queremos usar
- Mantener en una pestaña la referencia a Arduino que contenga las sentencias y funciones disponibles para la programación

Melanny Dávila

- Antes de implementar un diseño de un programa, se debe elegir el uso correcto de las librerías en base al tipo y modelo del sensor que se vaya a utilizar.
- Al momento de instalar una nueva librería se debe verificar que cumpla con los requerimientos con el fin de evitar que existan errores durante la ejecución del código.

REFERENCES

- [1] "IDE Arduino y Configuración," Aprendiendo Arduino, Jun. 18, 2017. [En línea]. Disponible en: <https://aprendiendoarduino.wordpress.com-2017-06-18-ide-arduino-y-configuracion/> (accedido Dic. 9, 2020).
- [2] E. Tatayo, "INTRODUCCIÓN A LA PROGRAMACIÓN EN ARDUINO Y TINKERCAD". C.P. SISTEMAS EMBEBIDOS, Accedido: nov. 22, 2020. [En línea].
- [3] O. Torrente, Arduino: Curso practico de formación, Alfaomega, 2013
- [4] J. G. Cobo, ¿Qué es Arduino?, Hardware libre, abr. 12, 2018. <https://www.hwlibre.com/que-es-arduino/> (accedido dic. 9, 2020).
- [5] C. Veloso, "Que es y para que sirve arduino - Ventajas de utilizar arduino", Electrónica analogica y digital, abr. 20, 2016. <https://www.electrontools.com/Home/WP/que-es-y-para-que-sirve-arduino/> (accedido dic. 09, 2020).
- [6] Jecrespom, "Uso de Librerías Arduino", Aprendiendo Arduino, jun. 20, 2017. <https://aprendiendoarduino.wordpress.com/2017/06/20/uso-de-librerias-arduino/> (accedido dic. 09, 2020).
- [7] Jecrespom, "Librerías Arduino", Aprendiendo Arduino, nov. 16, 2016. <https://aprendiendoarduino.wordpress.com/2016/11/16/librerias-arduino-2/> (accedido dic. 09, 2020).
- [8] "Arduino - GSM". <https://www.arduino.cc/en/Reference/GSM> (accedido dic. 09, 2020).
- [9] "ArduinoFFT - Open Music Labs Wiki". <http://wiki.openmusiclabs.com/wiki/ArduinoFFT> (accedido dic. 09, 2020).
- [10] "Arduino - AudioFrequencyMeter". <https://www.arduino.cc/en/Reference/AudioFrequencyMeter> (accedido dic. 09, 2020).
- [11] "Arduino - Ethernet". <https://www.arduino.cc/en/Reference/Ethernet> (accedido dic. 09, 2020).
- [12] A. Rapp, andrewrapp/xbec-arduino. 2020.