

Aguirre Prieto Ángel Ernesto, Castro Calvopiña Bryan Paúl y Ramos Espinosa
Christopher Lizardo

Resumen –en el presente proyecto el cual lo hemos realizado por varias partes se ha realizado una investigación a fondo de varias tarjetas de desarrollo las cuales nos mostraron varias facetas y en centramos varios de los beneficios, y sus componentes para así con la ayuda de este artículo agrandar nuestro conocimiento sobre los lenguajes de programación y las plataformas de simulación que pueden tener cualquiera de las siguientes tarjetas de desarrollo: Raspberry Pi, Arduino Uno, Micro Bit.

Índice de Términos – Microcontrolador: es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. **Interfaz:** Dispositivo capaz de transformar las señales generadas por un aparato en señales comprensibles por otro. **PWM:** pulse-width modulation. **AREF:** regulador de voltaje. **BLE:** Bluetooth Low Energy. **HDMI:** High-Definition Multimedia Interface. **Domótica:** Conjunto de técnicas orientadas a automatizar una vivienda, que integran la tecnología en los sistemas de seguridad, gestión energética, bienestar o comunicaciones.

Summary –in this project, which we have carried out by various parties, research has been carried out on a background of various development cards, which we are recognized as several facets and in contrast to various benefits, and their components, thus with the help of This article will increase our knowledge about the programming languages and the simulation platforms that any of the following development boards can have: Raspberry Pi, Arduino Uno, Micro Bit.

Keywords-Microcontroller: it is a programmable integrated circuit, capable of executing the orders recorded in its memory. **Interface:** Device capable of transforming the signals generated by one device into signals understandable by another. **PWM:** pulse-width modulation. **AREF:** voltage regulator. **BLE:** Bluetooth Low Energy. **HDMI:** High-Definition Multimedia Interface. Home automation: Set of techniques aimed at automating a home, which integrate technology into security systems, energy management, wellness or communications.

I. INTRODUCCIÓN

En este documento estudiaremos sobre las tarjetas de desarrollo en la cuales se encuentran la Raspberry Pi, Arduino Uno y la MicroBit se analizaran los campos de sus estructura y el uso de cada uno de los elementos se encuentran en cada una de ellas, también conocer los lenguajes de programación que se necesita para poder desarrollar alguna aplicación de las tarjetas de desarrollo además de conocer los simuladores proporcionados por el docente guía, en los mismos simuladores se realizaran ejemplos haciendo uso de lo investigado anteriormente como son los lenguajes de programación y las entradas de cada una de las tarjetas de desarrollo.

Para entrar en el campo del uso de las tarjetas de desarrollo se debe tener en cuenta la definición de las mismas la cual según varios expertos es: una plataforma de hardware de código abierto, basada en una placa de circuito impreso que contiene un microcontrolador que cuenta con entradas y salidas, analógicas y digitales, en un entorno de desarrollo que está basado en el lenguaje de programación. El dispositivo conecta el mundo físico con el mundo virtual, o el mundo analógico con el digital controlando, sensores, alarmas, sistemas de luces, motores, y actuadores.



II. DEFINICIÓN DE OBJETIVOS NECESARIOS PARA LA INVESTIGACIÓN

A. Objetivo general

Para definir el objetivo general se tiene en cuenta el problema

a estudiar y la búsqueda del producto solicitado después de realizar este proceso se logró definir el objetivo principal el cual es: Conocer las tarjetas de desarrollo y los elementos que la componen y sus funciones además de realizar ejemplos sobre los simuladores proporcionados para generar ejemplos funcionales de aplicaciones para las tarjetas de desarrollo.

B. Objetivos específicos

Cuando hablamos de definir los objetivos específicos tenemos que pensar que los mismos se descubren cuando se empiezan a hacer las respectivas investigaciones sobre el objetivo general mediante las cuales encontramos los siguientes objetivos específicos:

- Conocer los beneficios de usar las tarjetas de desarrollo.
- Conocer las partes que compone las tarjetas de desarrollo.
- Generar ejemplos funcionales en los simuladores de las tarjetas de desarrollo.
- Identificar la variedad de S.O disponibles para la instalación en las tarjetas de desarrollo.

II MARCO TEORICO

A. TAJETA DE DESARROLLO

Es una plataforma de hardware de código abierto, basada en una placa de circuito impreso que contiene un microcontrolador que cuenta con entradas y salidas, analógicas y digitales, en un entorno de desarrollo que está basado en el lenguaje de programación. El dispositivo conecta el mundo físico con el mundo virtual, o el mundo analógico con el digital controlando, sensores, alarmas, sistemas de luces, motores, y actuadores.

Las tarjetas de desarrollo están basadas en código abierto y software libre:

Código Abierto

El código abierto es un modelo de desarrollo de software basado en la colaboración abierta. Se enfoca más en los beneficios prácticos (acceso al código fuente) que en cuestiones éticas o de libertad que tanto se destacan en el software libre. Para muchos el término «libre» hace referencia al hecho de adquirir un software de manera gratuita.

El concepto también relaciona a una cultura que promueve el intercambio de ideas y valoriza la colaboración para mejorar el código de otros. El movimiento defiende su legado de colaboración, y ha producido una serie de manifiestos abogando por el desarrollo abierto de software.

Software Libre

Un software es libre si otorga a los usuarios de manera adecuada las denominadas cuatro libertades: libertad de usar, estudiar, distribuir y mejorar, de lo contrario no se trata de software libre. Existen diversos esquemas de distribución que no son libres, y si bien podemos distinguirlos sobre la base de

cuánto les falta para llegar a ser libres, su uso bien puede ser considerado contrario a la ética en todos los casos por igual.

El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que confundir al software libre con software gratuito (Freeware), que, conservando su carácter de libre, puede ser distribuido comercialmente "software comercial"

B. RASPBERRY PI

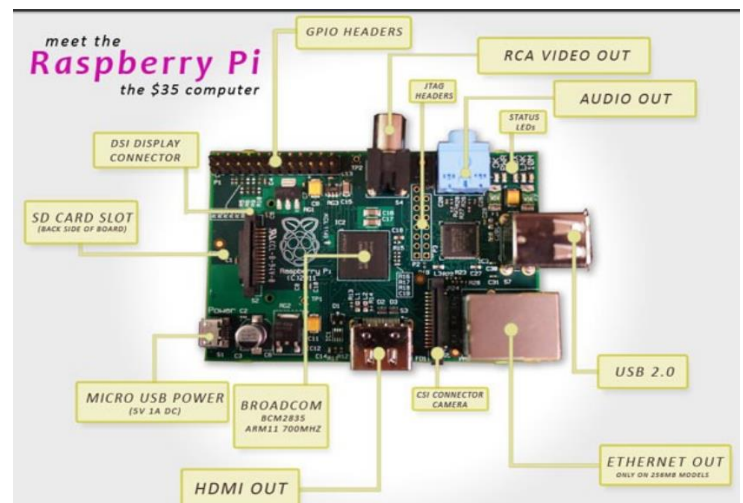
Raspberry PI es una placa computadora (SBC) de bajo coste, se podría decir que es un ordenador de tamaño reducido, del orden de una tarjeta de crédito, desarrollado en el Reino Unido por la Fundación Raspberry PI (Universidad de Cambridge) en 2011, con el objetivo de estimular la enseñanza de la informática.

Raspberry Pi surge con un objetivo en mente: Desarrollar el uso y entendimiento de los ordenadores en los niños. La idea es conseguir ordenadores portables y muy baratos que permitan a los niños usarlos sin miedo, abriendo su mentalidad y educándolos en la ética del “ábrelo y mira cómo funciona”.

No cabe duda que uno de los mejores leguajes utilizados para esta plataforma es PYTHON.

Una Raspberry pi contiene los siguientes elementos:

- SoC: Broadcom BCM2835 (CPU, GPU y SDRAM)
- CPU ARM1176JZF-S a 700 MHz de velocidad de reloj
- GPU Broadcom VideoCore IV
- Memoria RAM 512 MB (compartidos con la GPU)
- 2 x USB 2.0
- 1 x Salida audio mini jack 3.5 mm
- 1 x Salida audio/vídeo HDMI
- 1 x Salida vídeo compuesto RCA
- 1 x Micro USB
- 1 x RJ45 10/100 Ethernet RJ45
- Slot SD para tarjeta SD que contendrá el sistema operativo y programas Amazon Web Services (AWS)
- Alimentación: 5V/700 mA (3.5 W) vía micro USB



B. Artículo

En este paso se encuentra el juntar la información que antes se dividió con sus respectivas investigaciones para así conocer cada una de las características de cada una de la tarjetas de desarrollo además de conocer de manera teórica la función de cada uno de los pines y la razón por la cual se encuentran ubicados en las tarjetas de desarrollo.

En este paso se puede realizar varios de los primeros pasos sobre el artículo y el proyecto además de depurar los conocimientos y aumentarlos debido a la investigación a fondo realizada para cada uno de los elementos de las tarjetas.

C. Planteamiento de problema y objetivos

El planteamiento de los objetivos es esencial debido a que gracias a estos se puede llevar a cabo la realización del proyecto por tal motivo se definió los siguientes objetivos generales:

- Conocer las tarjetas de desarrollo y los elementos que la componen y sus funciones además de realizar ejemplos sobre los simuladores proporcionados para generar ejemplos funcionales de aplicaciones para las tarjetas de desarrollo.

Después de definir los objetivos generales se empezó a realizar las investigaciones respectivas y a raíz de lo surgieron los objetivos específicos que son los siguientes

- Conocer los beneficios de usar las tarjetas de desarrollo.
- Conocer las partes que componen las tarjetas de desarrollo.
- Generar ejemplos funcionales en los simuladores de las tarjetas de desarrollo.
- Identificar la variedad de S.O disponibles para la instalación en las tarjetas de desarrollo.

D. Planteamiento del estado del arte y el marco teórico

En este paso se realizó una investigación a fondo y en el marco teórico se colocó parte de los archivos más actuales de las cuales nos proporcionaron una mejor visión del panorama general además de proporcionarnos varias acotaciones a nuestro conocimiento.

E. MAPA DE VARIABLES Y PREREQUISITOS

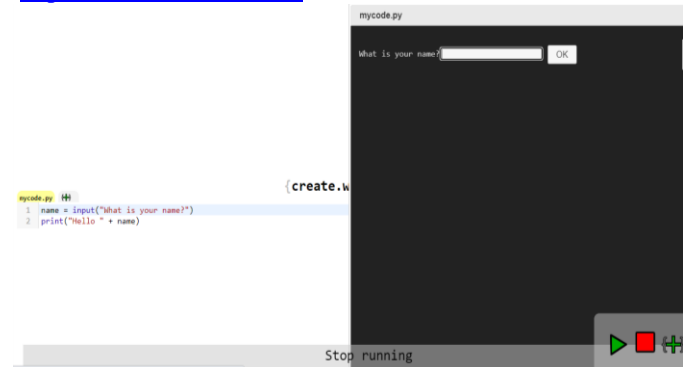
En esta parte de la investigación encontramos que las tarjetas de desarrollo tienen variaciones en específico la RASPBERRY tiene sus variaciones las cuales tienen cambios en sus pines y en la capacidad de procesamiento que tiene, otra variación que se encuentra es que algunas de ellas hacen uso de lenguajes de programación con algunas variaciones uno de estos casos es el de ARDUINO ya que hace uso del C++ modificado (simplificado).

F. DESARROLLO DE LOS EJEMPLOS DE CADA UNA DE LAS TARJETAS DE DESARROLLO

Ejemplo RASPBERRY PI

1- Identificar el simulador de la tarjeta en cuestión:
El simulador utilizado para esta tarjeta fue el Whitcode el cual se encuentra en el siguiente enlace:

<https://create.withcode.uk/>



2- Escribir el código en lenguaje de programación adecuado:

Se logró identificar en pasos anteriores que el lenguaje de programación en este caso es PYTHON

3- Ejecución del programa escrito en código:

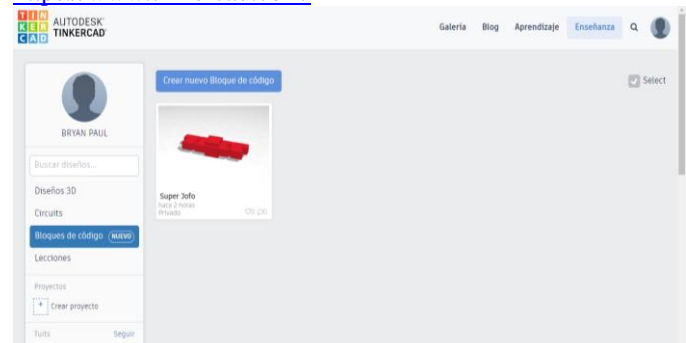
Para realizar la ejecución del código en el simulador solo es cuestión de dar click en el botón de acción play para que la misma empiece a correr.



ARDUINO UNO

1- Identificar el simulador de la tarjeta en cuestión:
El simulador utilizado para esta tarjeta fue el Tinkercad el cual se encuentra en el siguiente enlace:

<https://www.tinkercad.com/>

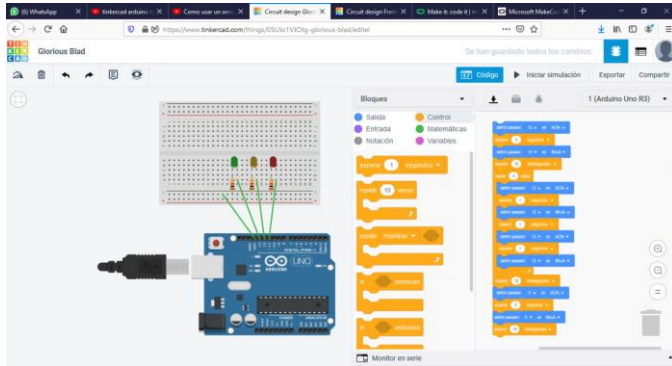


- 2- Escribir el código en lenguaje de programación adecuado:

Como ya se indicó en pasos anteriores el lenguaje de programación usado en esta tarjeta es el C++ (simplificado).

- 3- Ejecución del programa escrito en código:

En la plataforma proporcionada es muy fácil debido a que en la parte superior nos permite con un botón de acción correr la simulación y verla de manera gráfica gracias a que la plataforma es muy amigable.

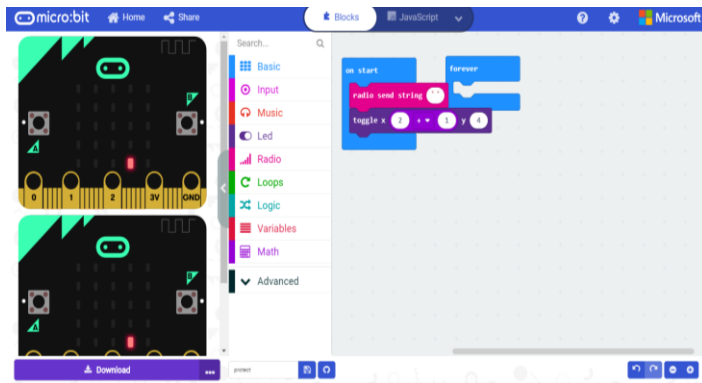


MICRO-BIT

- 1- Identificar el simulador de la tarjeta en cuestión:

El simulador a usar es el proporcionado ya que es el proporcionado por la página oficial del producto y se encuentra en este link:

<https://makecode.microbit.org/#editor>

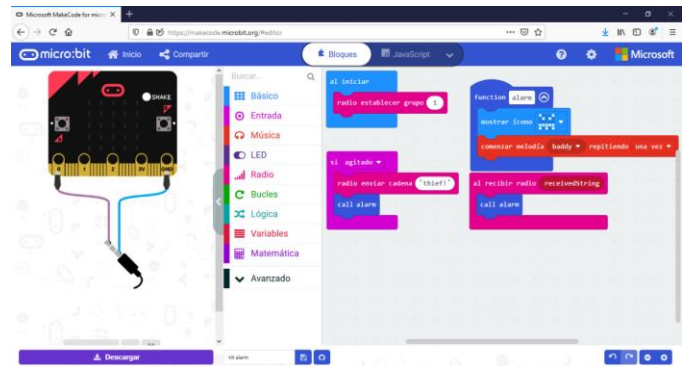


- 2- Escribir el código en lenguaje de programación adecuado:

Para escribir el código se lo pude hacer en JAVACRPT y PYTHON ya que es una tarjeta muy flexible.

- 3- Ejecución del programa escrito en código:

Para la ejecución del código y su simulación nos muestra en manera gráfica la placa y lo que sucedería cuando se corra el código escrito y este se ejecuta dando click en un botón de acción presente en la parte inferior de la imagen de la Tarjeta.



IV. CONCLUSIONES

Realizada la investigación sobre las tarjetas de desarrollo se llegó a las conclusiones sobre los objetivos específicos las cuales fueron:

- Se pudo comprender los distintos beneficios que trae el usar las distintas tarjetas ya que nos permiten conocer una manera más dinámica las partes y los usos ellos culés nos puede ayudar las mismas.
- Al realizar la investigación fue imperativo la busqueda de las funciones de las partes que componen las tarjetas sí que por tal motivo se tuvo una mejor comprensión y conocimiento de cada parte y acción de cada tarjeta.
- Gracias a los simuladores proporcionados por el docente se pudo simular y crear ejemplos funcionales sobre las tarjetas de programación ya que los simuladores proporcionados son amigables y bastante interactivos con el usuario.
- El repositorio que carece este servicio, es extenso, cumpliendo con todas las expectativas ante los sistemas operativos que se necesitan.
- Al realizar la investigación se nos mostro un aplico catálogo de sistemas operativos con los cuales las tarjetas pueden funcionar en su gran mayoría son los conocidos o en su caso varias distribuciones de GNU/Linux adaptados para las tarjetas de desarrollo.

Las conclusiones planteadas son de cada objetivo específico respectivamente los cuales ya fueron mostrados en un paso anterior.

Después de llegar a la conclusión de cada uno de los objetivos específicos se tiene que llegar a la evolución final del objetivo principal, que después de la evaluación se llegó a su respectiva conclusión, la cual es:

- Se logró con esto este objetivo dado que los objetivos específicos planteados nos llevaron paso a paso al cumplimiento del meso ya que por la investigación realizadas tiene un amplio conocimiento de las tarjetas de desarrollo y los elementos utilizados para las mismas además nos permitió la realizaciones ejemplos funcionales de entre los cuales ene te proyecto demuestran en la presentación.

V. RECOMENDACIONES

- Al momento de realizar la investigación nos mostró que se deben estudiar más lenguajes de programación debido a que hay plataformas en las cuales los lenguajes comunes no son suficiente y por ende nos genera problemas al rato de ejecutar y trabajar en otra plataforma a la acostumbrada.
- Conocer los distintos tipos de plataformas elegibles para realizar simulaciones y varias otras acciones para proyectos que posteriormente puedan ser solicitados.
- Tener en cuenta al momento de realizar cualquier programa sobre todo en programación que la sintaxis de las líneas de código sea correcta porque por el motivo se puede tener errores graves en el monto de ejecutar los programas creados.
- Para el uso del Control Remoto de Escritorio y si la maquina desde donde lo ejecutas no tiene habilitado el virtualizador, se deberá activar esta sección ya sea desde la BIOS o desde nuestro PANEL DE CONTROL, caso contrario no nos permitirá que genere imagen nuestra Virtual Machine.

VI. REFERENCIAS

- [1] JORGE CACHO HERNÁNDEZ, «Raspberry Pi: tutoriales Servidor web, ownCloud y XBMC.,» 27 Enero 2008. [En línea]. Available: <file:///C:/Users/home/Downloads/102190284-Raspberry-Pi-tutoriales-servidor-web-ownCloud-y-XBMC.pdf>
- [2] Microes, «Características y funcionalidades de MicroBit,» Microes, 15 septiembre 2017. [En línea]. Available: <http://microes.org/caracteristicas.php>
- [3] C. Muñoz, «Historia de la informática “Raspberry Pi,» Blog sobre Historia de la Informática, 18 Diciembre 2013. [En línea]. Available: <https://histinf.blogs.upv.es/2013/12/18/raspberry-pi/#:~:text=Raspberry%20PI%20es%20una%20placa,de%20la%20inform%C3%A1tica%20en%20las>
- [4] Maria Sol Vicet Illas, «Historia y defincion de software libre en el mundo e lainformatica inicial,» Ecured.cu, Ecuador, 2017 Available: https://www.ecured.cu/Software_libre#:~:text=Seg%C3%BAn%20la%20Free%20Software%20Foundation,programa%2C%20con%20cualquier%20prop%C3%B3sito%3B%20de
- [5] N. SATHISH, KUMAR; B. VUAYALAKSHMI; R. JENIFER PRARTHANA; A. SHANKAR, « IOT BASED SMART GARBAGE ALERT SYSTEM USING ARDUINO UNO,» IEE, 2019 Available: <https://ieeexplore.ieee.org/abstract/document/7848162>