

LEPERCOM - Una herramienta complementaria para la enseñanza de la programación

Martin Gustavo Juarez¹, Luis Roque Di Pinto²

¹ Departamento de Electricidad, Electrónica y Computación
FACET – Universidad Nacional de Tucumán
Tucumán, Argentina
mjuarez@herrera.unt.edu.ar

² Departamento de Electricidad, Electrónica y Computación
FACET – Universidad Nacional de Tucumán
Tucumán, Argentina
ldipinto@herrera.unt.edu.ar

Resumen. El presente trabajo busca introducir a los alumnos desde una edad temprana al concepto de la programación, ya sea programación de sistemas embebidos, microcontroladores o programación en general.

El trabajo se basa en el desarrollo de un sistema operativo diseñado para correr en un microcontrolador de bajos recursos con un costo menor a 1 U\$, que permite crear programas de usuario en diferentes lenguajes y plataformas, con la capacidad de realizar tareas en multitasking. El Sistema Operativo prevé la instalación de diferentes plugins para agregar funcionalidades y/o controladores para el manejo de diversos módulos y dispositivos de hardware existentes. El objetivo de este desarrollo, es lograr la mayor abstracción posible de las configuraciones requeridas por un microcontrolador haciendo fácil e intuitiva la programación.

Además, se desarrolló una capa de comunicación para facilitar la interconexión de dispositivos sin necesidad de configuración adicional por parte del usuario, permitiendo además la conexión a dispositivos como teléfonos y computadoras mediante Internet. Los dispositivos pueden trabajar colaborativamente y de manera descentralizada siguiendo la programación planteada por el usuario.

1 Introducción

Durante el año 2018 se desarrolló un Plan Nacional [1] que, entre sus propuestas, proponía incluir en la educación inicial núcleos de aprendizaje prioritarios de Educación digital, Programación y Robótica. Esta iniciativa fue destacada por la UNESCO en el documento "Artificial Intelligence in Education: Challenges and Opportunities for Sustainable Development" [2] ("Inteligencia Artificial en educación: desafíos y oportunidades para un desarrollo sostenible") donde se reconoce a Argentina como uno de los 5 países del mundo, por su política de preparación de los estudiantes para un futuro con IA (Inteligencia Artificial), orientado al desarrollo sostenible.

En función del nivel de educación, se podrían plantear diferentes objetivos acordes al nivel de preparación académica de los estudiantes, dividiéndolos en Nivel PRIMARIO, Nivel SECUNDARIO Técnico, y Nivel SECUNDARIO No Técnico.

1.1 Nivel Primario

Durante la educación primaria, los alumnos se capacitan en diferentes áreas cursando materias de índole variada. Esto ayuda a que cada niño encuentre su vocación e interés en el área que mejor se desempeñe, pero actualmente, las materias de programación no son enfatizadas suficientemente.

Resultaría de mucho interés que los estudiantes de nivel primario inicien su formación con un razonamiento lógico. Así podrán despertar el interés por la programación desde muy temprana edad, pudiendo forjar un razonamiento algorítmico.

1.2 Nivel Secundario

Dentro del nivel secundario podemos encontrarnos con dos líneas de educación muy diferentes. Aquellos alumnos que cursan un nivel secundario técnico, y aquellos que cursan un secundario social, natural o de otra área de conocimientos.

Independientemente de cuál sea la orientación de su educación secundaria en curso, los jóvenes de este nivel ya se encuentran acostumbrados a interactuar con diferentes dispositivos electrónicos. Por esto, su razonamiento lógico debe ser forjado desde otro punto de vista.

En búsqueda de proveer nuevas herramientas para facilitar la educación digital, se contempla el diseño colaborativo de dispositivos, tanto de hardware como de software. En particular, se desarrolló un Sistema Operativo para correr en una plataforma de bajo costo, la cual puede manejar diferentes dispositivos de diversos tamaños y funcionalidades, facilitando la configuración, la programación y el armado de una red interconectada.

El proyecto PIUNT 2023 “Internet de las cosas al alcance de todos: Estrategias Educativas mediados por tecnología” [3] propone el desarrollo de diversas herramientas educativas que utilizarán el Sistema operativo acá presentado, para apoyar la enseñanza de la programación en diferentes áreas.

2 Desarrollo

Al inicio se planteó crear una plataforma que permita fácilmente programar dispositivos interconectables entre sí, haciendo foco en su capacidad para enseñar los conceptos básicos de programación, permitiendo que el hardware, la comunicación y el funcionamiento multitarea sean lo más transparente posibles para el alumno.

Para ello se optó por desarrollar un Sistema Operativo (LeperOS) que se encargue de controlar todos los aspectos que deberían ser transparentes para el alumno.

El sistema operativo es el encargado de controlar y administrar el hardware del dispositivo sobre el que se ejecuta. En este caso, durante el proceso de diseño, se hizo

foco en que las configuraciones y la programación sean lo más fácil, intuitiva y transparente posibles para el usuario final.

Para facilitar la utilización de los periféricos disponibles y los programas del usuario, se unificó la nomenclatura con la cual se manipulan desde el sistema operativo. En términos generales, tanto programas, como periféricos y variables, representan para el sistema operativo un RECURSO disponible con el que se puede trabajar.

Los 5 aspectos más importantes que maneja el sistema operativo son:

- Entrada y salida (GPIO): administra los terminales de entrada y salida físicos del microcontrolador. Al momento de compilarse, el sistema operativo tiene en cuenta la información sobre qué microcontrolador se está usando y en qué placa está instalado (si fuera el caso). Con esta información, el sistema operativo lista los terminales en función de la serigrafía de la placa o indica el número de terminal del encapsulado, sin requerir documentación adicional o leer un diagrama que lo relacione a un componente de la placa, ni la documentación del micro que relacione el terminal físico con un puerto y bit.
- Administrador de Tareas: Para realizar diversas tareas en paralelo, LeperOS prevé la posibilidad de administrar la ejecución, no tan solo de varios programas, sino también de distintos tipos de programa. El usuario puede parar, pausar o ejecutar con distintas prioridades cada programa por separado. Asimismo, incluye un editor para la programación que permite al usuario crear los diferentes tipos de programas de manera simple y clara.
- Plugins: Dado que el usuario interactúa con el Sistema Operativo, la forma de agregar funcionalidades y/o controladores nuevos es mediante la instalación de plugins. Así, se puede elegir qué funcionalidades instalar en un dispositivo con memoria limitada. Los plugins pueden ser drivers para controlar hardware externo e inclusive paquetes de idioma.
- Comms: La herramienta LeperCom fue planteada para ser parte de una plataforma compuesta por varios dispositivos conectados entre sí. La comunicación entre ellos es un requisito del sistema operativo, y por esto una función importante es administrar las conexiones y protocolos de comunicación. Para ello se usan los puertos UART del microcontrolador utilizado.

El SO puede usar este puerto según la funcionalidad requerida:

- a. Conectar LeperComs entre sí para crear una red cableada de dispositivos que comparten sus respectivos recursos pudiendo cada LeperCom conectado a la red acceder a los recursos de los otros.
- b. Conectar un Gateway permitiendo a este dispositivo a conectarse a una red LAN / WAN, por donde conectarse a una RED con otros LeperComs. Además el LeperCom se puede conectar a una red cableada.
- c. Consola. Permite que un usuario se conecte mediante un terminal serie.
- d. Acceso a la aplicación del usuario permitiendo que el usuario controle el puerto desde alguna aplicación propia.

Como estos dispositivos prevén trabajar colaborativamente otra de las tareas del sistema operativo, es informar los valores actualizados de las variables propias que otros LeperComs de la red necesiten.

- Almacenamiento: El sistema operativo tiene la posibilidad de manejar tanto variables como programas. Todos estos deben ser alojados la memoria RAM disponible en el microcontrolador. A su vez, LeperOS se encarga de realizar copias en la memoria FLASH no volátil.

3 Resultados y Conclusiones

Las primeras pruebas con los dispositivos arrojaron resultados muy prometedores al poderse realizar las tareas de configuración y programación de manera rápida desde un entorno de consola. Si bien el entorno de consola no es conveniente para la mayoría de los usuarios, sirve para probar las bondades del sistema operativo.

Se realizaron pruebas con diferentes módulos de hardware y prototipos que permitieron realizar la configuración de periféricos tanto de entrada como de salida. También se pudo realizar el manejo de dichos periféricos para controlar luces y medir botones. Además, se realizó la programación de diferentes programas que relacionan dichos botones con luces mediante funciones lógicas. De esta manera se comprobó el funcionamiento de los programas y la velocidad de respuesta del sistema.

Finalmente, se realizaron pruebas de la comunicación entre dispositivos relacionando variables de uno, en función de las variables de otro que se encontraba conectado a la misma red. El resultado fue que las variables podían interactuar en función de los programas en cuestión, accediendo un dispositivo a las variables del otro sin problemas.

Podemos concluir que las pruebas realizadas resultaron muy satisfactorias y alientan a continuar con el desarrollo y la depuración del sistema operativo para poder implementarlo en la enseñanza de programación.

Referencias

1. ARGENTIA - Plan Nacional de Inteligencia Artificial - Ministerio de Educación, Cultura, Ciencia y Tecnología - (2018)
2. UNESCO (2019) ; Artificial Intelligence in Education: Challenges and Opportunities for Sustainable Development; disponible en: <https://unesdoc.unesco.org/ark:/48223/pf0000366994>
3. "Internet de las cosas al alcance de todos: Estrategias Educativas mediados por tecnología" - PIUNT 2023.
4. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)