





**INFOTEC CENTRO DE INVESTIGACIÓN E INNOVACIÓN  
EN TECNOLOGÍAS DE LA INFORMACIÓN Y  
COMUNICACIÓN**

DIRECCIÓN ADJUNTA DE INNOVACIÓN Y CONOCIMIENTO  
GERENCIA DE CAPITAL HUMANO  
POSGRADOS

**“NODO MULTIPLATAFORMA DE  
CONECTIVIDAD PARA IOT”**

IMPLEMENTACIÓN DE UN PROYECTO LABORAL

Que para obtener el grado de MAESTRO en Sistemas Embebidos

**Presenta:**

José Antonio Becerril Morales

**Asesores:**

Dra. Tania Aglaé Ramírez Del Real

Dr. Daniel Villanueva Vázquez

**Aguascalientes, Ags., Julio de 2020**

## Autorización de Impresión

## Agradecimientos

En primera instancia agradezco a mis padres, personas ejemplares, por darme la formación necesaria para crear la visión, constancia y perseverancia que me llevaron a culminar esta etapa de suma importancia en mi persona; ustedes son lo que más admiro. A mis hermanos por brindarme ese apoyo y empuje incondicional que me mantuvo motivado.

Mis asesores, que son formadores de grandes cualidades, gracias por transmitirme su sabiduría, esfuerzo y apoyo que llevaron a este punto mi trabajo de tesis. A mis maestros, ya que cada una de sus clases abrió mi mente para llegar al contenido de este documento.

A los lectores, espero que esta información le sea de utilidad, que encuentre lo que busca, y le ayude a clarificar o avanzar más rápido en sus objetivos.

Gracias a todos.

## Tabla de contenido

<b>Introducción</b> .....	<b>1</b>
<b>Capítulo 1. Antecedentes</b> .....	<b>3</b>
<b>1.1 Descripción del Problema</b> .....	<b>3</b>
<b>1.2 Motivación</b> .....	<b>4</b>
<b>1.3 Objetivos</b> .....	<b>7</b>
1.3.1 Objetivo General.....	8
1.3.2 Objetivos específicos .....	8
<b>1.4 Metodología de diseño</b> .....	<b>8</b>
<b>1.5 Estructura del documento</b> .....	<b>10</b>
<b>Capítulo 2. Marco teórico</b> .....	<b>11</b>
<b>2.1 Estado del arte</b> .....	<b>11</b>
<b>2.2 Computadoras Embebidas</b> .....	<b>12</b>
2.1.1 Microprocesador.....	14
<b>2.2 Arquitectura del sistema</b> .....	<b>15</b>
<b>2.3 Parámetros de selección de un microprocesador</b> .....	<b>15</b>
<b>2.4 Plataformas de conectividad para el Internet de las Cosas</b> .....	<b>17</b>
<b>2.5 Sistema operativo embebido</b> .....	<b>19</b>
<b>2.6 Gateway</b> .....	<b>20</b>
<b>2.7 Protocolos de comunicación</b> .....	<b>21</b>
2.2.1 WiFi .....	22
2.2.2 Bluetooth .....	23
2.2.3 USB.....	23
2.2.4 LoRa.....	23
<b>Capítulo 3. Desarrollo del proyecto</b> .....	<b>25</b>
<b>3.1 Diseño</b> .....	<b>25</b>
3.1.1 Requerimientos del sistema .....	25
3.1.2 Diagrama a bloques .....	27
3.1.3 Arquitectura del Nodo Multiplataforma de Conectividad para IoT. ....	28
<b>3.2 Selección de componentes</b> .....	<b>29</b>
3.2.1 Selección de microprocesador.....	29
3.2.2 Selección de Memoria ROM. ....	38
3.2.3 Selección de Memoria RAM.....	41
3.2.4 Selección de Módulo WiFi y Bluetooth.....	41
3.2.5 Selección de Módulo LoRa. ....	42
3.2.6 Selección de conectores USB. ....	42

3.2.7	Multiplexado de pines.....	43
<b>Capítulo 4. Implementación y resultados.....</b>		<b>47</b>
4.1	Proceso de inicio del sistema operativo embebido basado en linux.....	47
<b>4.2</b>	<b>Pruebas de comunicación.....</b>	<b>49</b>
4.2.1	Comunicación USB.....	50
4.2.2	Comunicación WiFi.....	51
4.2.3	Comunicación Bluetooth.....	53
4.2.4	Diseño de tarjeta LoRa.....	55
4.2.5	Diseño de hardware.....	56
4.2.6	Diseño de Software.....	63
<b>4.3</b>	<b>Resultados.....</b>	<b>69</b>
<b>Capítulo 5. Impacto Social y Económico.....</b>		<b>71</b>
5.1	Socios claves.....	71
5.2	Actividades clave.....	72
5.3	Recursos claves.....	73
5.4	Propuesta de valor.....	73
5.5	Clientes.....	74
5.6	Costos.....	74
<b>Conclusiones.....</b>		<b>76</b>
<b>Bibliografía.....</b>		<b>79</b>

## Índice de figuras

<i>Figura 1. Dispositivos conectados por persona en los últimos 17 años</i> .....	3
<i>Figura 2. Impacto económico de las áreas más importantes del IoT para el 2025</i> .....	4
<i>Figura 3. Metodología para el diseño del Nodo Multiplataforma</i> .....	8
<i>Figura 4. Diagrama básico de un microprocesador</i> .....	13
<i>Figura 5. Arquitectura base del Internet de las cosas</i> .....	22
<i>Figura 6. Diagrama a bloques del Nodo Multiplataforma de Conectividad para IoT</i> .....	27
<i>Figura 7. Arquitectura de Nodo Multiplataforma de Conectividad para IoT</i> .....	28
<i>Figura 8. Esquemático de configuración de reinicio de la tarjeta BeagleBone Black</i> .....	37
<i>Figura 9. Configuración de procesador a multiplexar</i> .....	44
<i>Figura 10. Configuración de multiplexado</i> .....	45
<i>Figura 11. Configuración de pines del procesador</i> .....	46
<i>Figura 12. Detalles de Hardware de BeagleBone Black</i> .....	49
<i>Figura 13. Configuración putty para conexión SSH de la BeagleBone Black</i> .....	50
<i>Figura 14. Inicio de sesión con putty</i> .....	51
<i>Figura 15. Conexión WiFi par BeagleBone Black</i> .....	52
<i>Figura 16. Pruebas de conexión WiFi</i> .....	53
<i>Figura 17. Configuración de controlador Bluetooth</i> .....	54
<i>Figura 18. Falla de conexión Bluetooth</i> .....	55
<i>Figura 19. Conexión exitosa bluetooth</i> .....	55
<i>Figura 20. Arquitectura de firmware del RN2483A</i> .....	56
<i>Figura 21. Esquemático propuesto por el fabricante del LDO</i> .....	58
<i>Figura 22. Esquemático del LDO</i> .....	60
<i>Figura 23. Esquemático del RN2483A</i> .....	61
<i>Figura 24. Esquemático de tira de pines de tarjeta LoRa</i> .....	62
<i>Figura 25. Layout de antenas LoRa</i> .....	63
<i>Figura 26. Imagen en 3D de la tarjeta LoRa</i> .....	63
<i>Figura 27. Pinout de tarjeta BeagleBone Black</i> .....	65
<i>Figura 28. Programa de comunicación LoRa</i> .....	66
<i>Figura 29. Diagrama de flujo de comunicación LoRa</i> .....	67

<i>Figura 30. Ejecución de programa de comunicación LoRa.....</i>	<i>68</i>
<i>Figura 31. Comandos de recepción de comunicación LoRa.....</i>	<i>68</i>
<i>Figura 32 Acoplamiento de tecnologías.....</i>	<i>69</i>



## Índice de cuadros

Cuadro 1 Características de una computadora embebida.....	13
Cuadro 2. Comparación de plataformas existentes. ....	18
Cuadro 3. Características Bluetooth.....	23
Cuadro 4. Requerimientos Funcionales.....	25
Cuadro 5. Requerimientos de Dominio.....	26
Cuadro 6. Requerimientos no funcionales.....	26
Cuadro 7. Comparación de microprocesadores.....	31
Cuadro 8. Descripción de los tipos de procesadores Cortex. ....	32
Cuadro 9. Características de los procesadores Cortex A7, 8, 9, y A64.....	33
Cuadro 10. Comparativa de costos entre procesadores. ....	33
Cuadro 11. configuración de pines SYSTEMBOOT.....	35
Cuadro 12. Configuración de los pines SYSTEMBOOT .....	36
Cuadro 13. Configuración de boot de la tarjeta BeagleBone Black.....	38
Cuadro 14. Descripción de tecnologías SD.....	39
Cuadro 15. Selección de conector microSD.....	40
Cuadro 16. Selección de memoria eMMC.....	41
Cuadro 17. Selección de memoria DDR3L.....	41
Cuadro 18. Selección de modulo Wifi/Bluetooth. ....	42
Cuadro 19. de selección de modulo LoRa.....	42
Cuadro 20. Selección de conectores USB.....	43
Cuadro 21. Costo de Procesador AM3358.....	43
Cuadro 22. Proceso de reinicio del procesador AM3358.....	48
Cuadro 23. Descripción de comandos para la configuración de módulo WiFi. ....	51
Cuadro 24. Descripción de comandos para la configuración de módulo Bluetooth. ....	54
Cuadro 25. Selección de regulador LDO.....	57
Cuadro 26. Selección de capacitores de desacoplo y resistencias de ajuste de voltaje....	59
Cuadro 27. Selección de LED, Resistencias y Botones Push .....	60
Cuadro 28. Tipos de comando de tranceiver RN2483A.....	64
Cuadro 29. Análisis de mercados por tecnología de comunicación del Nodo Multiplataforma. .....	71

*Cuadro 30. Recursos Clave para la fabricación del Nodo Multiplataforma. ....73*

*Cuadro 31. Costos de materiales del Nodo Multiplataforma.....75*

## Siglas y abreviaturas

**IoT:** Internet of Things

**RFID:** Radio Frequency Identification

**PCB:** Printed Circuit Boards

**SBC:** Single Board Computer

**GPU** Graphics processing Unit

**DDR:** Double Data Rate

**RAM:** Random Access Memory

**EMMC:** Embedded Multimedia Card

**EEPROM:** Electrically Erasable Programmable Read – Only Memory

**SD:** Secure Digital

**TI:** Texas Instruments

**HDMI:** High-Definition Multimedia Interface

**PCIe:** Peripheral Component Interconnect Express

**E/S:** Entradas y Salidas

**SPI:** Serial Peripheral Interface

**I2C:** Inter Integrated Circuits

**PWM:** Phase-Width modulation

**CAN:** Controller Area Network

**UART:** Universal Asynchronous Receiver - Transmitter

**GPIO:** General Purpose Input/output

**HMI:** Human Machine Interface

## Introducción

Actualmente el internet de las cosas (IoT, por sus siglas en inglés), está tomando mucho auge en la vida cotidiana del ser humano abarcando diferentes áreas, tales como energía, industria, ciudades inteligentes, agricultura y domótica. Este se ha convertido poco a poco en una necesidad para la integración de los dispositivos más comunes a redes alámbricas e inalámbricas del IoT; Estas redes están basadas en estándares tales como IEEE 802.15.4 (Instituto de ingenieros Eléctricos y Electrónicos, por su siglas en inglés)<sup>1</sup> el cual especifica la capa física y sus accesos a un dispositivo de bajo consumo de energía [1], y 6LowPAN (IPv6 Low-Power Wireless Personal Area Networks, por sus siglas en inglés) el cual es un protocolo de red que envía paquetes IPv6 en tramas pequeñas de la capa de enlace de datos como las definidas en IEEE 802.15.4.

La demanda de adaptar el IoT sobre dispositivos diversos, está creando la necesidad de diseñar Gateways o computadoras embebidas capaces de controlar y manipular las redes, así como su flujo de información de la capa física hasta la capa de servicios o mejor conocida como la nube.

En el presente trabajo se realiza la integración de hardware de un Gateway. Asimismo, se hace énfasis en las mejores prácticas de integración de hardware, en específico para el de cualquier computadora embebida.

Para el diseño se utiliza una metodología general que nos permita validar el hardware con el uso de una plataforma de conectividad ya existente, y que cuente con la mayoría de los protocolos de comunicación propuestos con la finalidad de optimizar tiempos de diseño, minimizar costos y errores de fabricación. Los protocolos no integrados en la plataforma se validan utilizando una tarjeta de propósito específico que permita entablar comunicación entre las mismas.

Finalmente, el uso y el diseño de Gateways como el “nodo multiplataforma de conectividad para IoT” es inminente para satisfacer la demanda actual de los

---

<sup>1</sup> <http://www.dinel.us.es/wie/?q=node/10>, IEEE WIE Spain, ¿Que es el IEEE?, diciembre 15 de 2019.

usuarios de dispositivos con conectividad. Por lo tanto, es necesario comenzar a adaptar la tecnología actual para la interacción que requiere el IoT entre sistemas y sobre todo para el manejo del flujo de información que se procesará por medio de estos dispositivos, lo cual es muy importante para la toma de decisiones.

Por último, se obtuvo como resultado una integración de hardware sobre los componentes de mayor impacto, que fue desarrollado hasta obtener su despliegue de componentes junto con sus costos a la fecha de diseño. También para probar el diseño propuesto se fabricó una tarjeta LoRa que nos permitiera generar una comunicación básica para validar el correcto funcionamiento del diseño.

El internet de las cosas se define como llevar conectividad y capacidad de cómputo a objetos de uso común o diario con la finalidad de obtener una comunicación entre los mismos [2]. Este conecta redes de dispositivos entre sí para establecer un flujo de información entre el medio físico y el internet, además se ha convertido en una tendencia creciente por los beneficios de conectividad y las comodidades que conlleva su implementación. Debido a esto nuevas tecnologías se integran a sus redes año con año facilitando la integración de nuevos dispositivos. Generalmente redes o sistemas del IoT se utilizan en ambientes donde se requiere automatizar la adquisición de información sin necesidad de interacción humana continua [3].



# Capítulo 1

## Antecedentes

# Capítulo 1. Antecedentes

## 1.1 Descripción del Problema

De acuerdo con el grupo de soluciones empresariales basadas en internet (IBSG, por sus siglas en inglés). El internet de las cosas surge en el momento que se conectan más dispositivos a internet que la cantidad de personas que existen en el planeta [4].



Figura 1. Dispositivos conectados por persona en los últimos 17 años

Fuente: D. Evans, 2011[4].

En la Figura 1 se muestra que a partir del año 2008 y 2009 existían más dispositivos conectados a internet que personas, es en este punto en el tiempo que el IBSG considera el surgimiento del internet de las cosas.

El problema que se presenta de acuerdo con la Figura 1 es de una saturación actual y futura de dispositivos conectados a internet y a redes privadas, ya sea institucionales, domésticas, comerciales o industriales, que a su vez nos lleva a un problema de acoplamiento, procesamiento y manipulación de arquitecturas, redes, ramas y mercados del internet de las cosas.

Con la evolución del diseño e implementación de sistemas IoT, se presenta la problemática de, cuando todos estos dispositivos estén integrados a una red de comunicación inteligente, como es que vamos a enlazarlos a una red de internet [5].

Además, como se menciona en el párrafo anterior, se ha llegado a una saturación en las redes que están demandando dispositivos que se encuentren en una capa de enlace que permitan la conexión entre el medio físico y el internet. Esto nos lleva a implementar una capa de aplicación capaz de controlar el tráfico de información de la red, esta capa está integrada por un Gateway.

## 1.2 Motivación

La implementación del internet de las cosas abarca varias áreas con un crecimiento tecnológico y económico prometedor proyectado para el año 2025, tal y como se plasma en la Figura 2 [6].

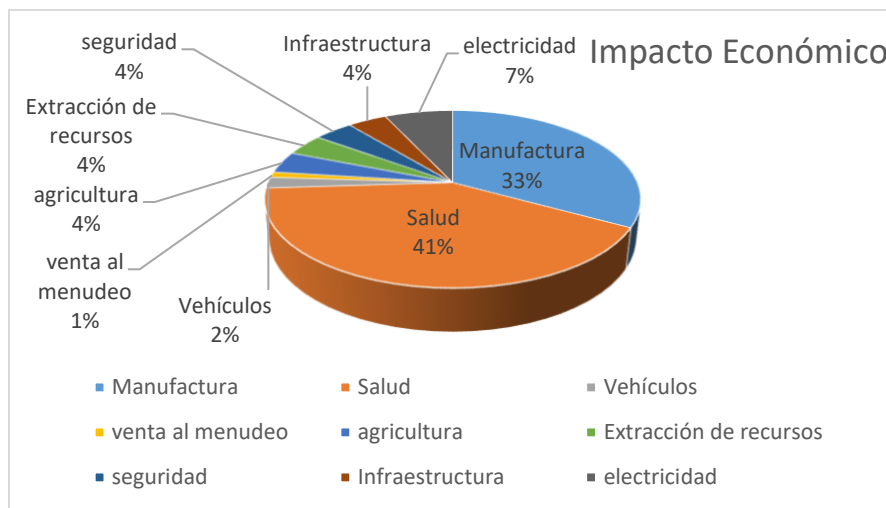


Figura 2. Impacto económico de las áreas más importantes del IoT para el 2025

Fuente: A. Al-Fuqaha, 2015[6].

Las ramas del IoT descritas en la Figura 2 son algunas de las más importantes y potenciales, pero no las únicas. Estas áreas cuentan con una arquitectura base que puede ser o no la misma, todo depende del uso e implementación del sistema, y aun cuando puede ser diferente su arquitectura todos los sistemas del IoT integran un Gateway, es decir que este es utilizado en todos los mercados para cada una de sus aplicaciones y redes.



El IoT en general se ha utilizado principalmente para la adquisición de información de manera continua para después procesarla y analizarla en la toma de decisiones a beneficio del usuario.

Mientras que el IoT se centra en productos individuales, en el área de manufactura su equivalente a implementar es la “Industria 4.0”, esta se centra en la producción industrial, y la diferencia principal entre estos está en la finalidad de uso, es decir, la finalidad del IoT es relacionar productos y dispositivos digitales con el internet hasta llegar al usuario, para la Industria 4.0 es enlazar máquinas, personas y productos para mejorar la producción y poder alcanzar una fábrica más eficiente e inteligente, el uso de la tecnología puede ser la misma para llegar a su fin [7]. Distinguiendo esto la industria 4.0 se puede implementar sensorizando maquinaria para recolectar información que ayude a conocer mejor a la misma, esto para mejorar procesos y predecir el estado actual de los sistemas [8]. En el caso del área de venta al menudeo la implementación del IoT llegó para revolucionar hacia el menudeo 4.0 o mejor conocido en inglés como retail 4.0 y se aplica principalmente para mejorar la experiencia del usuario en general, ya sea en compras presenciales o en línea, el punto es que el comprador no sienta diferencia alguna entre una y otra, y esto se hace posible integrando procesamiento de imágenes, identificación por radiofrecuencia (RFID, por sus siglas en inglés), Beacons (bluetooth de bajo consumo), y otras tecnologías inalámbricas que están siendo implementados para formar retail 4.0. La finalidad es la de crear un marketing que colecte datos de los usuarios para analizarlos y obtener información valiosa sobre los intereses de los compradores [9]. En el área de seguridad, se está trabajando en sistemas de video con grabaciones continuas, capaces de detectar eventos de interés para ser almacenados y analizados de manera rápida por el usuario. Esto permite desechar información redundante e innecesaria, la captación del evento se realiza por medio de la detección de rostros que se registra junto con la fecha y hora para poder ser recuperada cuando se requiera la identificación de un evento [10].

Para el caso del área de salud contamos con aplicaciones de prevención como dispositivos portables dotados con sensores de localización, caídas y cambios

significativos en uno o más de sus signos vitales. Esto con la finalidad de disparar alarmas que puedan ser recibidas para poder anticiparse a la situación [11]. El IoT aplicado a agricultura se conoce como “Agricultura de precisión” la cual permite el análisis de información en tiempo real para una mejor gestión de las decisiones. En este caso se desarrolla una plataforma que monitorea el medio ambiente del sistema por medio de sensores como humedad, temperatura, dirección del viento, nivel de pH y más. Esta información se envía a un Gateway que a su vez la envía a ser procesada para poder desplegar resultados en una aplicación de interacción humana [12]. Para el área de infraestructura una forma de mejorar la calidad de vida del ser humano es la implementación de semáforos inteligentes, el objetivo es controlar el flujo del tráfico de acuerdo con la congestión vehicular en determinado cruce o cruces, esto se puede lograr de diferentes maneras o con diferentes tecnologías, la cuestión es saber en determinado punto que tantos vehículos existen para poder tomar una decisión sobre el flujo de este. Algunos proyectos utilizan tecnología RFID, la cual permite la localización de los vehículos. Otros proyectos utilizan redes de sensores inteligentes, el cual está posicionado estratégicamente para proporcionar, posición, forma, y velocidad del vehículo. También existen semáforos controlados por medio de procesamiento de imágenes el cual detecta la densidad de tráfico comparando imágenes con tráfico denso y sin tráfico, de acuerdo con el resultado se determinan las luces y los tiempos de los semáforos del cruce [13]. Para el área vehicular se tienen proyectos de detección de ubicación y lectura de parámetros básicos para la medición del estado de un vehículo en cualquier recorrido, en este caso se utiliza la tecnología inalámbrica LoRa que nos permite comunicación de larga distancia con un consumo de energía bajo, la finalidad de este proyecto es la de prevenir o reaccionar a accidentes en viajes largos donde se puede llegar a perder la comunicación [14]. Para el área minera o extracción de recursos existen proyectos que atacan la problemática de la mala calidad del aire. El proyecto está adaptado para medir las condiciones del aire en minas de cielo abierto, se mide monóxido de carbono (CO), dióxido de nitrógeno (NO<sub>2</sub>), dióxido de azufre (SO<sub>2</sub>), y ozono (O<sub>3</sub>), material particulado, temperatura, presión barométrica y humedad relativa, y se utiliza un sistema embebido para

recaudar la información de cada sensor para poder transmitir su medición previamente procesada a una nube que permita su manipulación [15].

Para el área de electricidad se pretende resolver el problema de la medición de consumo energético en todo tipo de establecimiento, su finalidad es la de incrementar la eficiencia de cobro, así como controlar de manera óptima los cortes de energía, reactivaciones, y para reducir la contaminación mediante el uso óptimo de la energía. El principal objetivo es el de implementar un medidor de energía eléctrica inteligente que permita la comunicación en tiempo real con el sistema para conocer el consumo de determinado establecimiento [16].

El internet de las cosas se define como el intercambio autónomo de información entre dispositivos embebidos [17], es decir, que sin importar el área de aplicación la arquitectura del sistema es muy parecida en su implementación. Por lo tanto, en su mayoría integran un Gateway que funja como medio de comunicación, transporte de información y control de sus redes.

La implementación del Nodo Multiplataforma de Conectividad para IoT propone gestionar redes diseñadas por el usuario integradas por dispositivos u objetos del IoT.

La motivación del diseño del Nodo multiplataforma es que el IoT tiene la necesidad de interconectar billones o trillones de objetos a través de internet, por lo que es inminente la implementación de arquitecturas basadas en capas que integren un Gateway como el nodo multiplataforma. Y aunque el internet de las cosas está compuesto por sistemas con tecnologías cambiantes continuamente, todas tienen un Gateway en su arquitectura [6].

### **1.3 Objetivos**

En esta sección se presentan los objetivos, el general y los específicos.

### 1.3.1 Objetivo General

El objetivo general es el diseño de un Gateway para protocolos USB, WiFi, Bluetooth y LoRa, para ser integrado en sistemas del internet de las cosas.

### 1.3.2 Objetivos específicos

- a) Analizar el hardware esencial de una computadora embebida, para comparar plataformas ya existentes orientadas al IoT. Esto permite la selección de la plataforma ya existente para la validación de la integración de hardware del Nodo Multiplataforma.
- b) Seleccionar componentes esenciales del nodo multiplataforma.
- c) Analizar las mejores prácticas y recomendaciones de integración de hardware, para diseñar la tarjeta que contenga el protocolo LoRa que no soporta la plataforma seleccionada.
- d) Validar el funcionamiento de plataforma seleccionada y el funcionamiento de tarjeta de protocolos no contenidos.

## 1.4 Metodología de diseño

Se utilizó una metodología general basada en la metodología en cascada, que es un esquema secuencial de seguimiento de tareas [18].

Enfocada en la integración de hardware del Nodo Multiplataforma de Conectividad para IoT, quedaría como se muestra en la Figura 3.



*Figura 3. Metodología para el diseño del Nodo Multiplataforma.*

*Fuente: Elaboración propia.*

Esta metodología en cascada se enfoca en la aplicación de tareas secuenciales que permiten iteraciones cortas entre una y otra para minimizar errores. Ya aplicada al nodo, se forma por el análisis de requerimientos de usuario, para así poder transformarlos a los de diseño, los cuales son la directriz del desarrollo del proyecto. Una vez realizado este levantamiento se procede al análisis de requerimientos para proceder a su aplicación en el análisis de hardware. Este se realiza con una investigación sobre la integración de hardware de computadoras embebidas orientadas a ser un Gateway, integrando los protocolos USB, WiFi, Bluetooth y LoRa.

Desarrollados los bloques anteriores se realiza una investigación de análisis de plataformas existentes en el mercado, se procede a una comparativa ponderando la accesibilidad, su disposición en fuente abierta, y sobre todo la que más se acercan a los requerimientos del proyecto.

Esto con la finalidad de realizar un diseño basado en la plataforma seleccionada. Seguido de esto y de acuerdo con los requerimientos del proyecto se procede a la selección de componentes, principalmente el microprocesador. Y con base a las características técnicas de este, se realiza la selección de los componentes que forman el sistema mínimo del nodo multiplataforma, esto con el propósito de ofrecer el diseño y selección de los componentes críticos de la computadora embebida.

Ya con el sistema mínimo, se analizan y presentan buenas prácticas de integración de hardware para el correcto diseño de una computadora embebida, la cual es la base de hardware del nodo. Además, se diseña y manufactura una tarjeta que incluye los protocolos no soportados por la plataforma seleccionada.

El bloque final de validación y pruebas del sistema se realiza sobre la plataforma seleccionada para ahorrar tiempos y errores de diseño.

## **1.5 Estructura del documento**

El siguiente trabajo se estructura de la siguiente manera. En el capítulo 1, se realiza una introducción al internet de las cosas y la necesidad de la integración de Gateways para satisfacer la demanda del mercado. En el capítulo 2, se revisa el marco teórico del diseño de computadoras embebidas que se desempeñan como Gateways. En el capítulo 3, se desarrolla el proyecto con base a los requerimientos del sistema. En el capítulo 4, se plantean las recomendaciones de integración de hardware, el diseño de la tarjeta de protocolos no soportados por la plataforma existente, se revisan detalles de diseño y desarrollo. En el capítulo 5, se desarrollan las necesidades del proyecto para un lanzamiento rápido y exitoso al mercado.



## Capítulo 2

### Marco Teórico

## Capítulo 2. Marco teórico

### 2.1 Estado del arte

El internet de las cosas se encuentra en constante cambio ya que siempre está integrando nuevas tecnologías y protocolos de comunicación a sus sistemas. La tendencia actual del internet de las cosas es la de establecer comunicación entre dispositivos sin necesidad de tener un control externo o humano, por lo que es necesario dotar a los sistemas con Gateways de mayores capacidades de procesamiento que permitan lograr esta interacción [5]. Para la comunicación en el caso del Nodo se utiliza USB, uSD card, WiFi, Bluetooth, y LoRa.

Los Gateways se han innovado constantemente con la integración de protocolos de tendencias actuales, debido a esto nos encontramos con trabajos que adecuan el diseño del Gateway para trabajar con sistemas específicos para IoT. El trabajo presentado en [19] desarrolla e implementa un Gateway para recopilar información de uno o más invernaderos por medio de ZigBee, con la finalidad de enviar la información a un servidor para tomar decisiones y de controlar de manera remota, de igual manera tiene la facilidad de enviar mensajes de texto por GPRS. En [20] los autores realizan un Gateway basado en un smartphone, el cual implica utilizar los recursos del mismo para solucionar el problema de interoperabilidad que existe en el internet de las cosas, que es la comunicación dispositivo a dispositivo sin la necesidad de crear una plataforma de hardware específica. El smartphone ya cuenta con varios protocolos inalámbricos que evitan el uso de un hardware extra para el formado de redes, y se propone utilizarlo para entablar un tráfico de información y control. Por otro lado, AMBER [21] es una plataforma abierta enfocada a el uso de cualquier tipo de protocolo inalámbrico, esto lo logran agregando a su diseño de hardware 3 sockets llamados “Extender Module” los cuales son extensiones o sockets, en los cuales se pueden instalar diferentes tranceiver que contengan el protocolo necesario para el sistema a utilizar, la implementación de AMBER se enfoca a solucionar la modularidad, flexibilidad y escalabilidad en sistemas del internet de las cosas. Los autores en [22] enfocan el diseño de un



Gateway que solucione el problema de la concurrencia de datos manejada por medio de un protocolo de ordenamiento dinámico por prioridad.

La mayoría de los trabajos se centran en el diseño de un Gateway que resuelva el control y tráfico de información que presentan las redes de IoT, además todos contemplan protocolos de acuerdo con el sistema de IoT a implementar.

En este trabajo, además de resolver el tráfico, interoperabilidad y control de información, nos centramos en el diseño de la solución de hardware de una computadora embebida implementando una base de desarrollo que permita acelerar el diseño de Gateways basados en microprocesadores.

## **2.2 Computadoras Embebidas**

Las computadoras embebidas son plataformas de computación diseñadas para una tarea específica controlada por software. Las aplicaciones pueden variar desde automatización industrial, señalización digital, vehículos autónomos, hasta dispositivos de uso común. Existen varios tipos de computadoras embebidas, desde PC's industriales reforzados hasta PC's de panel, mini PC's, servidores industriales, computadoras para vehículos y Gateways IoT[23].

Las principales diferencias entre una computadora embebida y una computadora de escritorio son el propósito y el diseño.

CARACTERÍSTICA DE COMPUTADORA EMBEBIDA	OBSERVACIÓN
Equipo construido para realizar una tarea específica	La computadora embebida puede formar parte de un sistema más grande que realiza una tarea específica.
Pueden funcionar al máximo con pocos recursos	Su procesamiento puede funcionar en altos porcentajes aun cuando su sistema mínimo es limitado.
Soportar condiciones difíciles	Su diseño de hardware puede soportar condiciones climáticas extremas.
Tamaño reducido	Es una placa de tamaño reducido a especificación del usuario.
Altamente confiables en aplicaciones críticas	Al ser de uso dedicado sus recursos son enfocados en tareas críticas de manera eficiente.
Eficiencia energética	Su consumo de energía puede ser reducido y controlado.

Cuadro 1 Características de una computadora embebida.

Fuente: Elaboración propia.

Una computadora de placa reducida o computadora embebida, como se revisa en el cuadro 1, contiene todas las partes y características de un computador en una sola placa, es decir que contiene todas las prestaciones de hardware necesarias para su correcto funcionamiento [24]. Su organización es igual a la de una computadora de uso general la cual se compone de unidad central de procesamiento, sección de memoria, entradas y salidas. Estas secciones están interconectadas por tres conjuntos de líneas paralelas llamados buses que consisten en el bus de direcciones, bus de datos y bus de control [25], tal como se muestra en la Figura 4.

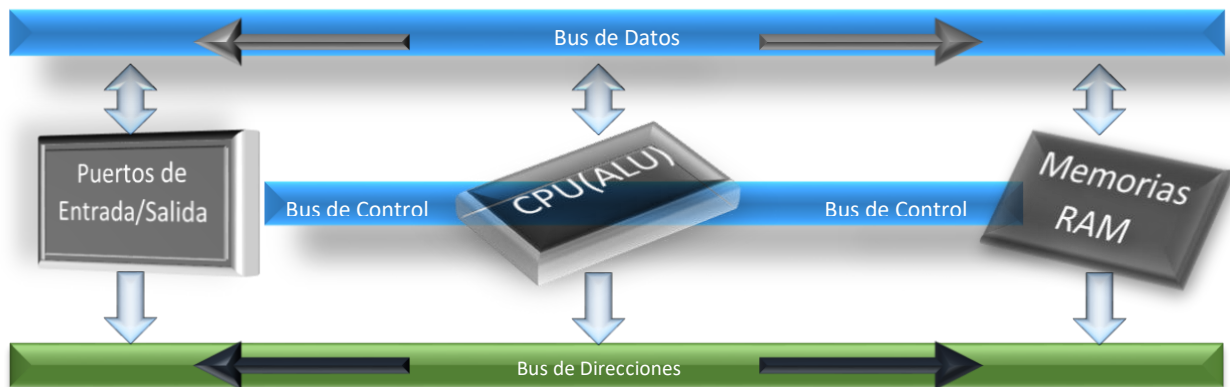


Figura 4. Diagrama básico de un microprocesador.

Fuente: Elaboración propia.

### **2.1.1 Microprocesador.**

El procesador es el encargado de la ejecución de instrucciones de programa [26], y se conforma por unidades interconectadas por diferentes circuitos externos que permiten su correcto funcionamiento.

El microprocesador unifica los circuitos integrados de un procesador en un solo empaquetado o circuito [25].

Tanto el procesador como el microprocesador cuentan con memoria interna, RAM y ROM, entradas y salidas limitadas, y los protocolos esenciales de comunicación como de almacenamiento para poder incrementar sus capacidades de memoria de manera externa. Ya que como se puede observar en la Figura 4 los microprocesadores cuentan con recursos limitados de manera interna.

El CPU controla las operaciones del sistema computarizado, este trae el código binario de la instrucción desde la memoria, decodifica las instrucciones a una serie de acciones simples y lleva acabo tales acciones [25]. La unidad aritmética lógica (ALU, por sus siglas en inglés) es una sección del CPU que realiza operaciones aritméticas y lógicas sobre datos llevados a esta. La ALU típicamente opera sobre uno o dos valores llamados operandos y los cambia acorde a un operador que se especifica por el fabricante [25].

Los puertos de entrada y salida permiten a la computadora recibir y enviar datos al mundo real [25]. Depende del diseño de cada microprocesador la cantidad y el tipo de puertos integrados en el mismo, la comunicación del CPU y los puertos es realizado por medio del bus de control. En el caso del Nodo Multiplataforma de Conectividad para IoT, tendremos el uso de los puertos de entrada y salida integrados en el microprocesador como protocolos de comunicación específicos a cada tecnología a utilizar.

De igual manera la unidad de memoria es accedida por medio del bus de control. Generalmente, esta unidad es una mezcla de memoria de acceso aleatorio (RAM, por sus siglas en inglés) y memoria de solo lectura (ROM, por sus siglas en inglés) [25]. Las unidades de memoria internas al microprocesador son limitadas,

admitiendo solamente la programación necesaria para realizar el arranque del microprocesador y tener accesos rápidos a la información, por lo que este bloque se conjuga con los puertos de entrada/salida para poder admitir memorias externas que permitan su integración al sistema y ofrecer una mayor flexibilidad [25].

## **2.2 Arquitectura del sistema**

La arquitectura del sistema se basa en la estructura de hardware del sistema, es decir, así como el microprocesador está integrado por un sistema mínimo para su correcto funcionamiento y arranque, el hardware externo de una computadora embebida está integrado por su propio sistema mínimo con base a los requerimientos del usuario para su correcto arranque y funcionamiento.

## **2.3 Parámetros de selección de un microprocesador**

La demanda de los dispositivos que cuentan con un microprocesador hace que estos se encuentren en constante mejora y evolución, con el tiempo incrementa el número de fabricantes, sus aplicaciones, y esto hace que constantemente cambien los criterios de selección.

Los criterios de selección del microprocesador para el Nodo Multiplataforma son los siguientes:

**Reloj del sistema:** Este es el motor del microprocesador, ya que trabaja por medio de un cuarzo que al aplicarle voltaje este convierte la energía eléctrica en un movimiento mecánico oscilatorio que provoca a su vez una oscilación de energía a su salida. Esta oscilación llega hasta el circuito PLL que es el encargado de convertirla en secuencias de pulsos digitales. El reloj del sistema trabaja con la frecuencia de estos pulsos que es un factor cuantitativo que indica la velocidad del microprocesador [27].

Las instrucciones máquina que acepta el microprocesador consumen un ciclo de reloj, así cuanto mayor sea la frecuencia del chip, menor será el tiempo que se

tardará en ejecutar las instrucciones de un programa, con esto podemos determinar un mejor rendimiento del microprocesador [27]. Se debe tomar a consideración que, a mayor frecuencia mayor consumo de energía, por lo tanto, su disipación de temperatura aumenta.

**Memoria cache:** La memoria caché es una memoria de accesos rápidos situada entre la memoria RAM y el microprocesador. En esta se almacenan los datos o códigos de programa que el microprocesador utiliza con mayor frecuencia. El controlador caché se encarga de intuir los datos que son más probables que el microprocesador utilice. Estos datos se extraen de la memoria RAM del sistema y los lleva a la memoria caché antes de que el procesador los requiera [28]. La memoria caché se constituye de memoria estática en lugar de dinámica con tiempos de acceso inferiores a 1 nano segundo, esto regularmente se relaciona a la eficiencia del controlador de caché, sin embargo, este depende en gran medida de la eficiencia del sistema. Por esto, se debe considerar que la relación entre el tamaño de la memoria y el aumento de la eficiencia no es lineal, ya que el aumento de memoria caché ayuda a eficientar los accesos a RAM, más no los accesos que realiza el sistema a la misma. Por este motivo se recurre a escalar la memoria caché creando niveles que se acoplan en cascada de forma que van aumentando su tamaño según se van separando del núcleo de la CPU [28].

**Número de núcleos:** La tecnología de núcleo múltiple evitó seguir aumentando la frecuencia en los microprocesadores para tener mejores prestaciones. Con núcleos múltiples se puede ejecutar más de una instrucción al mismo tiempo de programas distintos, la cantidad de instrucciones a ejecutar en paralelo va a depender del número de núcleos que integren el microprocesador [26].

**Arquitectura:** La arquitectura o estructura interna del microprocesador es de suma importancia, ya que esta puede ser creada para diferentes aplicaciones, es decir, que las prestaciones de un microprocesador no necesariamente mejorarán la eficiencia de este para cualquier aplicación, sino que son arquitecturas específicamente creadas para ciertas aplicaciones. Esta característica de selección

es de las más importantes ya que puedes tener diferentes combinaciones de las anteriores para llegar a la selección del microprocesador adecuado [26].

## 2.4 Plataformas de conectividad para el Internet de las Cosas

Actualmente existen muchas plataformas de hardware o computadoras embebidas dedicadas a la conectividad del internet de las cosas. Se evaluaron algunas de estas para la realización del hardware del Nodo Multiplataforma.

Estas son de hardware abierto, es decir, que proporcionan toda su información de fabricación para mandar manufacturar sus diseños, copiarlo o basarse en el mismo para beneficio del usuario. En el caso de este trabajo se realizó una comparativa de las tarjetas con más información y disponibilidad en el mercado para poder basar nuestro diseño en la misma. Esto nos ayuda a reducir costos y tiempo de lanzamiento al mercado.

CARACTERÍSTICA	BEAGLEBONE BLACK WIRELESS	OPEN REX	ORANGE PI PC PLUS	BANANA PI -M64
PROCESADOR	AM3358 ARM CORTEX-A8	IMX6 ARM CORTEX-A9	ALLWINNER H3 ARM CORTEX-A7	ALLWINNER A64 ARM CORTEX A64
NUCLEOS	1	4	4	4
VELOCIDAD DE RELOJ	1 GHZ	1.2 GHZ	1.6 GHZ	1.2 GHZ
GPU	SGX 3D GRAPHICS ENGINE	GPU 3D VIVANTE GC2000 GPU 2D (VECTOR GRAPHICS) VIVANTE GC355 GPU 2D(COMPOSICIÓN) VIVANTE GC320	ARM MALI400MP2	MALI-400 MP2 @ 500MHZ
MICRO CONTROLADOR	ARM CORTEX A8	ARM CORTEX M3 72 MHZ	NO	NO
MEMORIA INTERNA	512 MB DDR3800 MHZ RAM	RAM 272KB	32 KB L1 CACHE DE INSTRUCCIONES Y 32KB L1 CACHE DE DATOS POR CORE 512KB L2 CACHE	32 KB L1 CACHE DE INSTRUCCIONES 32KB L1 CACHE DE DATOS POR CORE 512KB L2 CACHE
MEMORIA EXTERNA	4GB 8-BIT EMMC MICROSD	DDR3-1066 UP TO 4GB MICROSD EEPROM 16KBIT I2C FLASH 32 MBIT SPI	1GB DDR3 EMMC FLASH-8GB	2GB DDR2 SDRAM MICROSD CARD EMMC 8GB

CARACTERÍSTICA	BEAGLEBONE BLACK WIRELESS	OPEN REX	ORANGE PI PC PLUS	BANANA PI -M64
COMPATIBILIDAD DE SOFTWARE	DEBIAN, ANDROID, UBUNTU, OPEN SOURCE LINUX KERNEL	YOCTO, ANDROID	ANDROID, LUBUNTU, DEBIAN, RASPBERRY PI IMAGE	ANDROID Y LINUX
WIFI	TI WILINK-802.11B/G/N	NO	SI	802.11 B/G/N (AP6212)
ETHERNET	NO	10/100/1000 MBPS ETHERNET	10/100M ETHERNET J45	10/100/1000MBPS ETHERNET (REALTEK RTL8211E/D)
BLUETOOTH	BLUETOOTH 4.1 Y BLUETOOTH LE	NO	NO	BLUETOOTH 4.0 (AP6212)
USB	HS USB 2.0 CLIENT TIPO-A MINI HS USB2.0 HOST TIPO-A	USB OTG MICRO 2 USB	USB 2.0 HOST USB 2.0 OTG	USB 2.0 USB OTR (MICRO USB)
SALIDA VIDEO	HDMI TYPE-D (MICROHDMI)	HDMI	HDMI	HDMI
ENTRADA VIDEO	NO	ENTRADA CSI, ENTRADA DIFERENCIAL PARA CAMARAS	ENTRADA CSI	ENTRADA CSI
SALIDA AUDIO	NO	AURICULARES	HDMI	JACK Y HDMI
ENTRADA DE AUDIO	NO	MIC	MIC	MIC
PCIE	NO	MINI SLOT	NO	NO
E/S	65 DIGITAL I/O, SPI, I2C, PWM, CAN, UART, SD, GPIOs 7 ENTRADAS ANALÓGICAS	I2C, CAN, UART, SPI, PWM, GPIOs, 4 ENTRADAS ANALÓGICAS	UART, SPI, CAN, PWM	GPIO, UART, I2C, I2S, SPI, PWM
CONSUMO DE ENERGÍA	210-460 MA. @ 5V	SIN INFORMACIÓN	SIN INFORMACIÓN	SIN INFORMACIÓN
APLICACIÓN	AUTOMATIZACIÓN CONECTIVIDAD Y APLICACIONES IOT	INDUSTRIAL Y AUTOMATIZACIÓN, EL CPU PUEDE SER UTILIZADO PARA: HMI, TABLETS, DISPOSITIVOS MÉDICOS PORTABLES, CONECTIVIDAD VEHÍCULO A VEHÍCULO, SISTEMAS DE AUDIO Y MÁS.	CONECTIVIDAD Y MULTIMEDIA	CONECTIVIDAD Y MULTIMEDIA

Cuadro 2. Comparación de plataformas existentes.

Fuente: AM335x Sitara Processors, 2014[29], F. Semiconductor, 2012[30], M.A. Processor, 2015[31].

Las tarjetas del cuadro 2 fueron seleccionadas debido a su disponibilidad, costo y facilidad de información.

Una característica que estas tarjetas comparten es que ninguna cuenta con el protocolo de comunicación LoRa integrado. Debido a esto en el capítulo 4, se

realiza el diseño de una tarjeta de circuito impreso LoRa para satisfacer las necesidades de este proyecto.

## 2.5 Sistema operativo embebido

El sistema operativo es el software que en la mayoría de las veces se ejecuta en modo kernel, y realiza dos funciones básicas, proporcionar a los programadores de aplicaciones con un conjunto abstracto de recursos simples, en vez de los complejos de hardware; y administrar estos recursos. Cabe destacar que más del 90% de las CPU's en el mundo se encuentran en sistemas embebidos, y la mayoría de ellos utilizan procesadores de 32 y 64 bits que ejecutan un sistema operativo completo [32].

Típicamente los sistemas embebidos complejos llevan un sistema operativo, aunque en teoría según los principios de la computadora es posible utilizar uno de uso general, como un Linux, sin embargo para sistemas embebidos, tenemos limitaciones de memoria, consumo de energía y los requerimientos de tiempo real dictan el uso de un sistema operativo diseñado para estos ambientes [33]. En el caso del nodo multiplataforma de conectividad para IoT, cuenta con el procesador AM3358, el cual es compatible con sistemas operativos embebidos de alto nivel, como lo son todos los basados en linux y Android [29]. Los microprocesadores solo pueden ejecutar código que existe en sus memorias internas ROM o RAM; los sistemas operativos por su tamaño residen en sistemas de memoria con mayor capacidad que la interna de un procesador, por ejemplo, un sistema operativo para la tarjeta BeagleBone Black pesa alrededor de 4GB<sup>2</sup>, y debe ser de capacidad permanente. Cuando el procesador es encendido su memoria interna no contiene el sistema operativo, por lo que es necesario un software especial para llevar el

---

<sup>2</sup> <https://beagleboard.org/latest-images>, BeagleBoard.org, Latest Images, diciembre 15 de 2019



sistema operativo a la memoria destino. Este software es normalmente pequeño de código y es llamado bootlaoder [34].

En un sistema embebido el role de un bootloader es complicado ya que este es raro que contenga un BIOS para realizar las configuraciones iniciales, y aunque las inicializaciones de bajo nivel varían de tarjeta a tarjeta y de CPU a CPU, las configuraciones se deben realizar antes de ejecutar el sistema operativo [34].

## **2.6 Gateway**

Un Gateway es un dispositivo denominado puerta de enlace porque es capaz de unificar diferentes protocolos de comunicación y arquitecturas en un solo sistema [5]. El flujo de información que maneja va de una capa física a una capa de alto nivel, de servicio o de usuario, donde regularmente ocurre el procesamiento de datos y la toma de decisiones [35]. Su propósito principal es el manejo de las conexiones de red de su sistema y el tráfico de datos de una capa a otra. Esto implica una división importante que impacta en la arquitectura del sistema como en la complejidad de hardware del Gateway. Esta división es el manejo de la conectividad de la red, el procesamiento de información y la interfaz de usuario. Lo más recomendable es dividir estas etapas para reducir el costo de hardware, dando a la capa física del mismo, solamente el manejo de la conectividad entre protocolos y el control del flujo de información entre la capa física y la de usuario, de esta manera las restricciones de hardware disminuyen. El análisis de información se debe implementar en la nube o capa de servicios en donde se cuenta con mayor poder de procesamiento [5].

En el caso del nodo multiplataforma de conectividad para IoT realiza esta separación adaptando el hardware con los protocolos WiFi, Bluetooth, USB y LoRa.

## 2.7 Protocolos de comunicación

La selección de los protocolos de comunicación de un Gateway define la arquitectura de su sistema, una arquitectura base para sistemas de IoT está dividida en la capa de redes o capa física, la capa de procesamiento y la capa de usuario, de aquí podemos determinar que un Gateway debe contar con al menos dos protocolos, uno para el armado y control de redes, y otro para enviar la información a un servidor o capa de usuario.

El internet de las cosas se encuentra en un crecimiento constante demandando diferentes tipos de sistemas, tecnologías y topologías, debido a esto podemos encontrar arquitecturas generales que nos ayudan a identificar a estos. Una de las arquitecturas generales consiste en tres capas, aplicación, transporte, y sensado [36].

**Capa de aplicación:** implementa tecnologías con un nivel de procesamiento alto que nos ayude a obtener información valiosa de los datos procesados, además ofrece una interface entre los usuarios y el sistema [36].

**Capa de transporte:** Es la encargada de la administración del tráfico de información y del control de las redes de sensado [36]. Esta capa es en la que se encuentran los Gateways.

**Capa de sensado:** Es la responsable de adquirir la información del medio físico [36].

De esta arquitectura podemos comenzar para complementar y resolver todos los problemas del IoT. Una mejora rápida y muy importante es la de agregar una capa de seguridad.

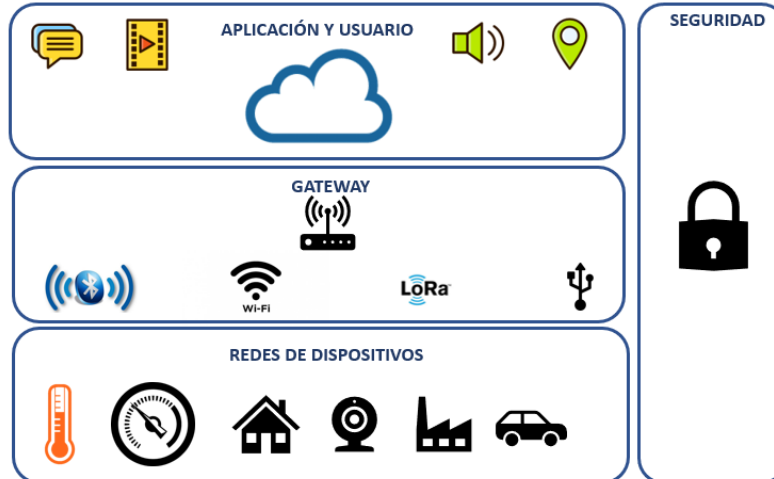


Figura 5. Arquitectura base del Internet de las cosas.

Fuente: E.L.E. Ieee[36].

### 2.2.1 WiFi

Anteriormente las soluciones inalámbricas se basaban en “Wireless Local Area Network” o WLAN de manera específica para cada fabricante, es decir, que cada uno manipulaba la capa de hardware o física a conveniencia. Esto obligaba a los usuarios a adquirir la tecnología o solución de cada fabricante sin opción a una interoperabilidad entre los mismos. El IEEE dio solución a esto implementando el estándar IEEE 802.11 para redes inalámbricas [37].

Cuando se habla de la tecnología WiFi, en realidad se hace referencia a la WiFi Alliance, una organización sin fines de lucro, que engloba a un amplio grupo de fabricantes con el objetivo de promocionar el uso de la tecnología inalámbrica en redes de área local y asegurando la compatibilidad de fabricantes con base a los estándares IEEE 802.11 [38]. La expansión de este tipo de tecnología ha sido explosiva y se prevé que en los próximos 1 o 2 años el 90% de los equipos ya dispongan de WiFi [39]. Este protocolo se utiliza en el Nodo como la interfaz de comunicación entre la nube y el Gateway o capa física.

### 2.2.2 Bluetooth

La tecnología inalámbrica bluetooth es un estándar de comunicaciones cuyo principal objetivo es eliminar los cables entre dispositivos conectados a su red, bluetooth es una especificación global y abierta especialmente diseñada para la conexión inalámbrica de voz y datos a corta distancia [40].

BLUETOOTH	
ALCANCE EN MTS	10 -100
CAPACIDAD DE TRANSMISIÓN	720 KBPS
FRECUENCIA	2400 - 2483 MHZ
TRANSMISION DE VOZ	CANALES SÍNCRONO A 64Kbps
TRANSMISION DE DATOS	CANALES ASÍNCRONOS DE 721 Kbps UN SENTIDO y 57.6 KBPS EN EL OTRO, O ENLACE SIMÉTRICO A 432KBPS EN AMBOS SENTIDOS

*Cuadro 3. Características Bluetooth.*

*Fuente: A. Rodríguez Penin, 2009[40].*

En la Cuadro 3 se describen las características de trabajo del protocolo Bluetooth, donde podemos observar que este está diseñado para distancias cortas y para transmisiones de datos no críticos, o de respuesta lenta.

### 2.2.3 USB

“Universal serial bus”, por sus siglas en inglés USB que como su nombre lo dice es un protocolo de comunicación en serie entre dispositivos electrónicos. Este cuenta con una capacidad de interacción de 127 dispositivos o periféricos como máximo [41]. Esta tecnología fue diseñada para utilizarse como entrada/salida.

### 2.2.4 LoRa

“Longe Range”, por sus siglas en inglés LoRa, es una capa de hardware que utiliza la modulación de radio frecuencia para establecer comunicación de largo alcance. Su tecnología está basada en la técnica de modulación “chirp spread spectrum”, por sus siglas en inglés CSS, esta técnica esta implementada en la capa física de LoRa y transmite en la banda de 868MHz a largas distancias [42]. CSS es una técnica de

propagación de espectros por medio de pulsos llamados “chirp” estos pulsos son señales de frecuencia modulados para codificar la información [43]

CSS tiene las mismas características de potencia que “Frequency shifting Keiyng”, por sus siglas en inglés FSK, pero a diferencia de esta incrementa el rango de alcance de comunicación. CSS ha sido utilizada en comunicaciones espaciales y militares debido a su largo alcance y su resistencia a interferencia y LoRa es la primera implementación de CSS de bajo costo comercial. Las ventajas de LoRa están en la tecnología de largo alcance, ya que un transceiver LoRa puede cubrir ciudades enteras o cientos de kilómetros cuadrados [44].

LoRa es la capa física que se utiliza para crear las redes, solo se enfoca en el control de hardware para su correcta ejecución de transmisión o recepción. Debido a esto LoRa Alliance crea LoRaWAN (Low Power Wide Area Network, por sus siglas en inglés) el cual es un protocolo diseñado para la conexión y comunicación entre dispositivos inalámbricos de bajo consumo de energía.

La topología o la arquitectura de una red LoRaWAN está basada en un despliegue de Gateways en estrella que a su vez despliega sus dispositivos de la misma manera. En esta topología el Gateway mantiene la comunicación entre sus dispositivos controlados y la interfaz de usuario a través de su servidor o nube. En nuestro caso el Nodo Multiplataforma de Conectividad para IoT fungiría de esta manera en este tipo de despliegues<sup>3</sup>.

---

<sup>3</sup> <https://lora-alliance.org/about-lorawan>, LoRa Alliance, What is LoRa WAN Specification?, Julio 22 de 2020.



## Capítulo 3

# Desarrollo del proyecto

## Capítulo 3. Desarrollo del proyecto

### 3.1 Diseño

El diseño del Nodo está basado en una metodología general o en cascada, revisada previamente en la sección 1.4, esta nos permite desarrollar los pasos consecutivos necesarios para poder ofrecer una integración de hardware estable y de rápido lanzamiento.

#### 3.1.1 Requerimientos del sistema

Los requerimientos son exigencias del usuario para el diseño y desarrollo de un sistema, estos ayudan al diseñador a clarificar lo que el usuario demanda o requiere del sistema a diseñar. Además, al ser condiciones o necesidades estas se transforman en métricas para una correcta planeación y diseño del proyecto.

Los requerimientos se pueden dividir en tres tipos, funcionales, no funcionales, y de dominio [45]. En el cuadro 4 se definen los requerimientos funcionales, los cuales describen las capacidades del sistema en un nivel alto o como lo dice su nombre “funcional”.

CÓDIGO	ESPECIFICACIÓN
<i>RF-01</i>	EL SISTEMA DEBE INICIAR AL SER CONECTADO A LA FUENTE DE PODER
<i>RF-02</i>	EL SISTEMA DEBE ENCENDER AL PRESIONAR UN BOTÓN
<i>RF-03</i>	EL SISTEMA DEBE REINICIAR AL PRESIONAR UN BOTÓN
<i>RF-04</i>	EL SISTEMA DEBE COMUNICARSE POR MEDIO DE WIFI, LORA, BLUETOOTH, Y USB.
<i>RF-05</i>	EL SISTEMA CUENTA CON MEMORIA RAM, ROM Y SD EXTERNA.
<i>RF-06</i>	EL SISTEMA PUEDE CARGAR EL SISTEMA OPERATIVO POR MEDIO DE LA MEMORIA SD Y ROM.

*Cuadro 4. Requerimientos Funcionales.*

*Fuente: Elaboración propia.*

En el cuadro 5 se define las funcionalidades del sistema con restricciones del dominio, es decir, todo lo que el sistema puede realizar, pero con restricciones ya específicas.

CÓDIGO	ESPECIFICACIÓN
<b>RD-01</b>	EL SISTEMA DEBE TENER UN CONECTOR DE ENTRADA DE VOLTAJE/ALIMENTACIÓN DE 3V A 5V
<b>RD-02</b>	EL SISTEMA DEBE TENER UN CONECTOR USB QUE PROPORCIONE ALIMENTACIÓN/VOLTAJE
<b>RD-03</b>	EL SISTEMA DEBE TENER BOTÓN DE ENCENDIDO
<b>RD-04</b>	EL SISTEMA DEBE TENER BOTÓN DE REINICIO.
<b>RD-05</b>	EL SISTEMA DEBE CONTENER UN TRANCEIVER WIFI
<b>RD-06</b>	EL SISTEMA DEBE CONTENER UN TRANCEIVER LORA
<b>RD-07</b>	EL SISTEMA DEBE CONTENER UN TRANCEIVER BLUETOOTH
<b>RD-08</b>	EL SISTEMA DEBE CONTENER UN PUERTO USB
<b>RD-09</b>	EL SISTEMA DEBE CONTENER UNA MEMORIA RAM EXTERNA DE AL MENOS 2GB
<b>RD-10</b>	EL SISTEMA DEBE TENER UNA MEMORIA ROM EXTERNA DE AL MENOS 8GB
<b>RD-11</b>	EL SISTEMA DEBE TENER RANURA PARA MEMORIA SD

*Cuadro 5. Requerimientos de Dominio.*

*Fuente: Elaboración propia.*

En el cuadro 6 se definen los requerimientos no funcionales, que son los que no impactan en la funcionalidad del producto, pero si en las características medibles de uso.

CÓDIGO	ESPECIFICACIÓN
<b>RNF-01</b>	EL SISTEMA CUENTA CON INDICADOR DE ENCENDIDO
<b>RNF-02</b>	EL SISTEMA CUENTA CON EXTENSIÓN DE PINES DIRECCIONADA A LOS DEL PROCESADOR.

*Cuadro 6. Requerimientos no funcionales.*

*Fuente: Elaboración propia.*



### 3.1.2 Diagrama a bloques

De acuerdo con los requerimientos se obtiene el diagrama a bloques del sistema, el cual se muestra en la Figura 5.

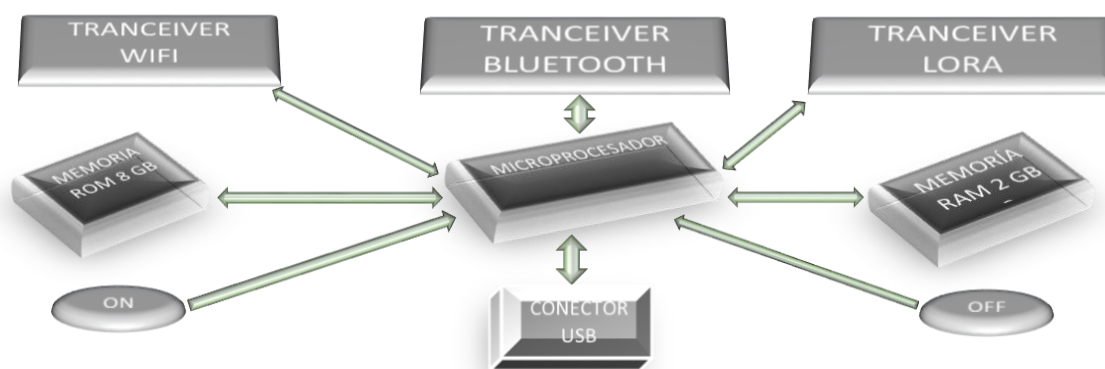


Figura 6. Diagrama a bloques del Nodo Multiplataforma de Conectividad para IoT.

Fuente: Elaboración propia.

El diagrama a bloques indica los componentes de mayor impacto en la solución de hardware. Como se puede observar en la Figura 5, el componente central es el microprocesador, ya que su selección determina el hardware externo, es decir, las características de funcionamiento y los protocolos de cada uno de los componentes electrónicos a utilizar. Conocer lo que se necesita del sistema de acuerdo con los requerimientos, determina las capacidades o características del microprocesador, asimismo, esto nos permite seleccionar la plataforma existente para poder basar nuestro diseño, además ofrece las entradas y salidas para poder plantear la arquitectura del sistema.

### 3.1.3 Arquitectura del Nodo Multiplataforma de Conectividad para IoT.

El Nodo multiplataforma funge como Gateway en cualquier sistema de IoT, la ventaja que este tiene es la de contener los protocolos más utilizados en las redes del internet de las cosas, pudiendo configurar los mismos para el manejo de cualquier red de dispositivos compatible.

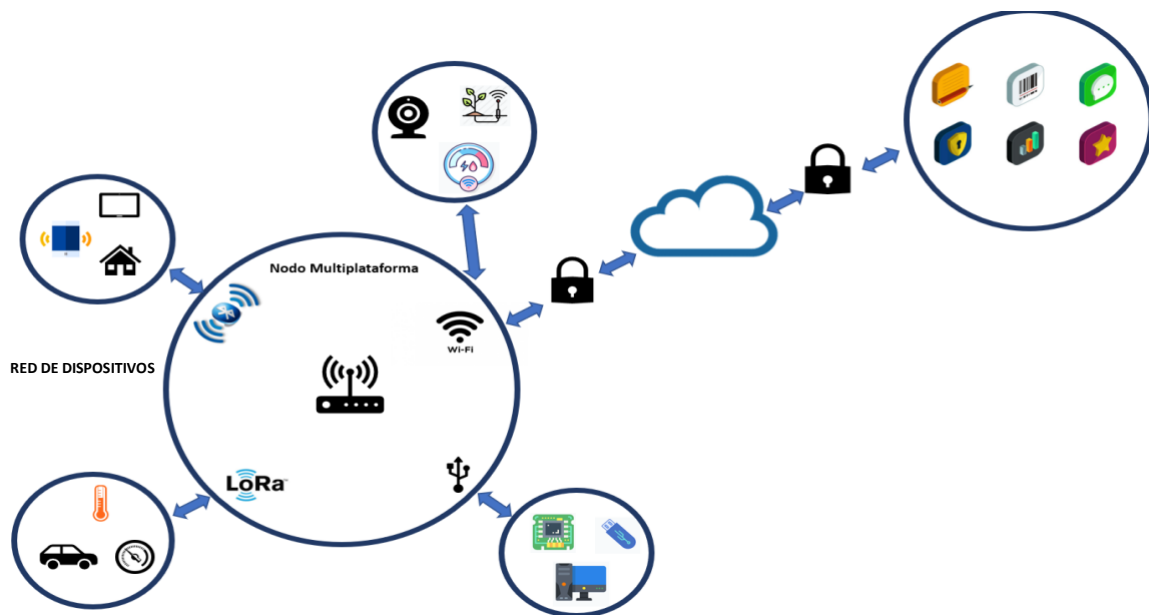


Figura 7. Arquitectura de Nodo Multiplataforma de Conectividad para IoT.

Fuente: Elaboración propia.

Esta arquitectura es de uso general del Nodo Multiplataforma de Conectividad para IoT, sin embargo, esta se puede adaptar dependiendo del sistema y área de aplicación, la flexibilidad del IoT y la versatilidad del Nodo permiten crear diferentes topologías de red a preferencia del usuario.

**Red de dispositivos:** Es la capa que interactúa directamente con el mundo real, está formada por una o más dispositivos de acuerdo con su protocolo, además cada uno puede tener diferentes topologías de red [46].

**Nodo Multiplataforma de Conectividad:** Es la capa donde se localizan los Gateways, este funge como enlace de control e información entre las redes y la capa de aplicación [46].

**Capa de Seguridad:** Es la capa que analiza la información, del usuario hacia el Gateway y viceversa, asegurando confidencialidad, autenticidad y confianza entre la información compartida y del sistema entero [46].

**Capa de Aplicación:** es la capa que permite procesar la información recibida para poder presentarla en la capa de interfaz [46].

**Capa de interfaz de usuario:** es la capa de presentación de información al usuario, además en esta capa es posible manipular el sistema entero a consideración de este [46].

En los sistemas que involucra internet, específicamente hablando del IoT, están expuestos siempre a permitir accesos indebidos a sus redes y diferentes capas de arquitectura, estos podrían tener acceso a dominios administrativos, de usuario e incluso de interacción con su medio ambiente, por lo que se recomienda no excluir la capa de seguridad.

## **3.2 Selección de componentes**

La tecnología por seleccionar depende del microprocesador, ya que su arquitectura dicta protocolos de comunicación, niveles eléctricos, características y diseño de la tarjeta y más. Esto quiere decir que en base a los requerimientos técnicos del procesador se seleccionan los componentes externos para que cumplan con los mismos y esto permita un correcto diseño y acoplamiento de hardware.

### **3.2.1 Selección de microprocesador**

Un microprocesador cuenta con varias características en su arquitectura interna y externa que determinan su eficiencia, sin embargo,

su desempeño sobre la aplicación es la característica que prepondera su selección, debido a que su diseño es espacialmente acoplado para esa tarea en específico, por lo que se espera una mayor eficiencia.

La selección se realizó con base a los parámetros definidos en el capítulo 2 en la sección 2.3. que son:

- **Reloj del sistema**, frecuencia de trabajo del sistema.
- **Memoria Cache**, capacidades y organización.
- **Número de núcleos**, con cuantos núcleos cuenta el procesador.
- **Arquitectura**, diseño distribución interna para la ejecución de una aplicación en específico.

Es importante destacar que estas características generalizan el rendimiento de un microprocesador, sin embargo, para poder hacer válida una comparación entre procesadores tendrá que ser de la misma familia, y dependiendo de su arquitectura podría hacerse entre fabricantes también.

De acuerdo con nuestro estudio de mercado de las plataformas más adecuadas para la implementación del Nodo Multiplataforma de Conectividad para IoT, en el cuadro 7 tenemos los procesadores con sus características de selección.

PLATAFORMA	PROCESADOR	CORES	FRECUENCIA	ORGANIZACIÓN DE MEMORIA
<b>BEAGLEBONE BLACK WIRELESS</b>	AM3358 ARM CORTEX-A8	1	1 GHZ	* 32KB DE L1 INSTRUCCIONES Y 32KB DE DATOS *256KB DE L2 *176KB DE BOOT ROM *64KB DE RAM *64KB L3 RAM *SOPORTA MDDR, DDR2, DDR3, DDR3L, 16BITS DE BUS DE DATOS *SOPORTA MMC, SD, SDIO
<b>OPEN REX</b>	IMX6 ARM CORTEX- A9	4	1.2 GHZ	* 32KB DE L1 INSTRUCCIONES Y 32KB DE DATOS *1MB DE L2 *96KB DE BOOT ROM *64KB DE RAM *SOPORTA DDR3, DDR3L, LPDDR2-800,16,32 Y 64 BITS DE BUS DE DATOS *SOPORTA MMC, SD, SDIO

PLATAFORMA	PROCESADOR	CORES	FRECUENCIA	ORGANIZACIÓN DE MEMORIA
<b>ORANGE PI PC PLUS</b>	ALLWINNER H3 ARM CORTEX-A7	4	1.6 GHZ	* 32KB DE L1 INSTRUCCIONES Y 32KB DE DATOS *512KB DE L2 *96KB DE BOOT ROM *SOPORTA DDR2, DDR3, DDR3L, LPDDR2, LPDDR3, 32 BITS DE BUS DE DATOS *SOPORTA MMC, SD, SDIO
<b>BANANA PI -M64</b>	ALLWINNER A64 ARM CORTEX A64	4	1.2 GHZ	* 32KB DE L1 INSTRUCCIONES Y 32KB DE DATOS *512KB DE L2 *112KB DE BOOT ROM *SOPORTA DDR2, DDR3, DDR3L, LPDDR2, LPDDR3, 32 BITS DE BUS DE DATOS *SOPORTA MMC, SD

Cuadro 7. Comparación de microprocesadores.

Fuente: AM335x Sitara Processors 1, 2014[29], F. Semiconductor, 2012[30], M.A. Processor, 2015[31], Allwinner, 2007[47].

Todos los procesadores a evaluar cuentan con arquitectura ARM la cual es una arquitectura RISC por sus siglas en inglés “Reduced Instruction Set Computer” la cual contiene las siguientes características [48]:

1. Registros de archivo de gran tamaño
2. Arquitectura de carga/almacena, donde las operaciones de datos procesados se realizan en los registros y no en memoria.
3. Modos de direccionamiento simples, donde el direccionamiento de carga/descarga se determina de contenido de los registros y los campos de instrucciones.
4. Campos de instrucciones uniformes y de gran tamaño, para simplificar la decodificación de instrucciones.

Además la arquitectura ARM provee adicionalmente [48]:

1. Control sobre la ALU y su controlador de cambios en casi todos los procesamientos de datos en sus instrucciones para maximizar el uso de estos.
2. Auto incremento y decremento de los modos de direccionamiento para optimizar bucles de programas.
3. Carga y almacenamiento de múltiples instrucciones para maximizar el rendimiento de datos.

- Ejecución condicional de casi todas las instrucciones para maximizar el rendimiento de ejecución.

ARM es solo la arquitectura y CORTEX es el procesador que aplica la misma en su hardware, cortex se diferencia dependiendo de la aplicación y la versión de arquitectura empleada en el mismo [49].

PROCESADOR	DESCRIPCIÓN
CORTEX-A	ES USADO PARA APLICACIONES DE TIEMPO REAL CON UN ALTO RENDIMIENTO, COMO IMPRESORAS, CONTROLADORES DE DISCO DURO, Y BOLSAS DE AIRE EN AUTOMÓVILES.
CORTEX-R	ES USADO PARA APLICACIONES DE TIEMPO REAL CON UN ALTO RENDIMIENTO, COMO IMPRESORAS, CONTROLADORES DE DISCO DURO, Y BOLSAS DE AIRE EN AUTOMÓVILES. ESTE ES MUY SIMILAR A UN MICROCONTROLADOR DE GAMA ALTA, PERO SU OBJETIVO SON SISTEMAS MÁS GRANDES DE LOS QUE NORMALMENTE SE USAN EN UNA UNIDAD DE CONTROL NORMAL.
CORTEX-M	ESTE FUE DISEÑADO ESPECÍFICAMENTE PARA EL MERCADO DE LOS MICROCONTROLADORES, ESTE PUEDE SER IMPLEMENTADO COMO CORE EN UN FPGA, PERO ES MUCHO MÁS COMÚN ENCONTRARLOS EN UN MICROCONTROLADOR, ALGUNOS SON OPTIMIZADOS PARA EFICIENCIA ENERGÉTICA, ALTO RENDIMIENTO Y ALGUNOS SON FABRICADOS ESPECIALMENTE PARA MERCADOS COMO SMART METERING.

Cuadro 8. Descripción de los tipos de procesadores Cortex.

Fuente: Silicon Labs, 2020[50].

De acuerdo con el cuadro 8, los microprocesadores que se encuentran dentro de nuestra comparación son del tipo Cortex-A, en específico A7, A8, A9, y A64. Los cortex A5 al A17 son parte del primer rango de procesadores cortex y están basados en la arquitectura ARMv7-A.

PROCESADOR	DESCRIPCIÓN
CORTEX-A7	ES UNO DE LOS CORES CON MENOR CONSUMO DE ENERGÍA E INCREMENTO DE EFICIENCIA CONTRA EL A5, ADEMÁS ES COMPATIBLE CON EL A15 Y A17. ESTE ES IDEAL PARA APLICACIONES DE SMARTPHONES Y TABLETS EN CUESTIÓN DE AHORRO MONETARIO. EN GENERAL ES UN CORE DE ALTA EFICIENCIA ENERGÉTICA Y PUEDE SER APLICADO EN SISTEMAS QUE REQUIERAN UN BALANCE ENTRE CONSUMO DE ENERGÍA Y EFICIENCIA <sup>4</sup> .
CORTEX-A8	ES EL PROCESADOR PERFECTO PARA INTEGRARLO EN DISEÑOS DE ALTO RENDIMIENTO EN PRODUCTOS FINALES, YA QUE TIENE LA CAPACIDAD DE SER CONFIGURADO A ALTAS

<sup>4</sup> <https://developer.arm.com/ip-products/processors/cortex-a/cortex-a7>. Arm Limited, Cortex-A7 Technical Reference Manual, diciembre 15 de 2019

PROCESADOR	DESCRIPCIÓN
	FRECUENCIAS. CONTIENE CARACTERÍSTICAS QUE PERMITEN APLICAR PROCESAMIENTOS ACELERADOS DE MULTIMEDIA Y PROCESAMIENTO DE SEÑALES <sup>5</sup> .
<b>CORTEX-A9</b>	ES UNO DE LOS CORES MÁS UTILIZADOS Y MADUROS DE ARM, ES ESPECIALMENTE UTILIZADO EN SISTEMAS DE BAJO CONSUMO DE POTENCIA Y DE PRESUPUESTO LIMITADO, Y EN DISPOSITIVOS DE 32 BITS QUE REQUIEREN DE UN DESEMPEÑO COMPETITIVO <sup>6</sup> .
<b>CORTEX-A64</b>	EL A64 ESTÁ BASADO EN EL PROCESADOR A53 DE ALTA EFICIENCIA QUE IMPLEMENTA LA ARQUITECTURA ARMV8-A. ESTE TIENE UNA MUY ALTA EFICIENCIA EN EL CONSUMO DE ENERGÍA A COMPARACIÓN DE SUS PREDECESORES, TIENE MAYOR RENDIMIENTO QUE EL A7 EL CUAL ES BASTANTE UTILIZADO EN APLICACIONES MÓVILES <sup>7</sup> .

Cuadro 9. Características de los procesadores Cortex A7, 8, 9, y A64.

Fuente: Silicon Labs, 2002[50].

A pesar de que todos los procesadores son cortex – A, cada uno de ellos está diseñado para solucionar diferentes problemas en sus mercados objetivos.

PLATAFORMA	PROCESADOR	COSTO DIGIKEY	OBSERVACIONES
<b>BEAGLEBONE BLACK WIRELESS</b>	AM3358 ARM Cortex-A8	\$17.21	CARACTERÍSTICAS PROPUESTAS
<b>OPEN REX</b>	iMX6 ARM Cortex-A9	\$28.23	SOLO SE ENCUENTRA DISPONIBLE A FRECUENCIA DE 1GHZ
<b>ORANGE PI PC PLUS</b>	Allwinner H3 ARM Cortex-A7	N/A	NO ENCONTRADO
<b>BANANA PI -M64</b>	Allwinner A64 ARM Cortex A64	N/A	NO ENCONTRADO

Cuadro 10. Comparativa de costos entre procesadores.

Fuente: Elaboración propia.

Los criterios de búsqueda en la página de DigiKey, se realizaron de acuerdo con las características presentadas en el “cuadro 2 Comparación de plataformas existentes”. Uno de los criterios más importantes fue el de la disponibilidad de compra, por lo que los cortex A7 y A63 fueron descartados. En el caso del Cortex-A9 solo se encontró a una frecuencia de 1GHZ, que es la misma frecuencia del A8. A pesar de que el procesador cortex A9 es ideal para el desarrollo de dispositivos

<sup>5</sup> <https://developer.arm.com/ip-products/processors/cortex-a/cortex-a8>, Arm Limited, Cortex-A8 Technical Reference Manual, diciembre 15 de 2019

<sup>6</sup> <https://developer.arm.com/ip-products/processors/cortex-a/cortex-a9>, Arm Limited, Cortex-A9 Technical Reference Manual, diciembre 15 de 2019

<sup>7</sup> <https://developer.arm.com/ip-products/processors/cortex-a/cortex-a53>, Arm Limited, Cortex-A53 Technical Reference Manual, diciembre 15 de 2019

por su alta eficiencia energética y de procesamiento, el A8 está diseñado para proporcionar un alto rendimiento en dispositivos finales, además, una de sus funciones principales es la de ser integrado en dispositivos para IoT como el Nodo Multiplataforma. Y de acuerdo con el cuadro 11, el costo del A8 es más bajo a la compra de una pieza a comparación del A9.

Es importante aclarar que en el desarrollo del presente trabajo se está enfocando el diseño a la reducción de costos y tiempo de diseño, y el AM3358 se encuentra empotrado en una de las placas de fuente abierta o mejor conocido como “Open source en inglés”, contiene bastante información de referencia que ayuda a acelerar el diseño basado en el procesador AM3358, por lo que es elegido para ser utilizado en el Nodo Multiplataforma.

Como se menciona anteriormente la base de diseño de la placa es el microprocesador, sus características o capacidades de hardware dictaminan los componentes a seleccionar.

El procesador AM3358 de Texas instruments cuenta con una guía de diseño de hardware que cubre los puntos más importantes a realizar para el diseño de sistemas basados con este procesador, sin embargo estos puntos se pueden acoplar a diseños basados en diferentes procesadores con las recomendaciones de hardware del fabricante pertinentes [51].

Los pasos para cubrir son:

**Diseño de diagrama a bloques del sistema**, el cual debe contener todos los componentes de hardware de alto nivel [51]. Este punto se cubrió anteriormente en el diseño de diagrama a bloques de la Figura 5.

**Seleccionar el modo de reinicio o mejor conocido en inglés como “Boot”** [51].

Los procesadores AM3358 cuentan con un bootloader interno grabado en ROM que nos permite cargar el sistema operativo a las memorias externas. Este procesador soporta el reinicio por medio de NOR Flash, NAND Flash, SPI, SD/MMC, EMAC, UART, y USB.



Estos modos de Boot son seleccionables a consideración del diseñador o de acuerdo con los requerimientos del sistema, en el caso de este trabajo debemos cumplir con el requerimiento funcional “RF-06” el cual demanda que el sistema debe reiniciarse o cargar el sistema operativo por medio de la memoria SD o ROM externas.

De acuerdo con este requerimiento el microprocesador cuenta con un procedimiento de Boot implementado por hardware y software que nos permiten realizar la configuración solicitada.

El bootloader interno del procesador realiza los siguientes pasos para poder determinar la carga del sistema operativo:

- a) Al energizar el procesador corre el bootloader interno que contiene una subrutina que crea una lista de dispositivos basada en la configuración de reinicio por software o por hardware. En el caso de hardware la configuración se realiza al conectar a voltaje o tierra los pines “SYSTEMBOOT” [52]. El SYSTEMBOOT cuenta con 15 pines descritos como sigue:

SYSTEMBOOT PINS	DESCRIPCIÓN	OBSERVACIÓN
15 AL 14	FRECUENCIA DEL CRISTAL	DE 19.2MHZ A 26MHZ
13 AL 12	APLICA PARA TODOS LOS TIPOS DE BOOT	APLICAR 00 PARA OPERACIÓN NORMAL
11 AL 10	PARA XIP Y NAND	XIP: DISPOSITIVO MULTIPLEXADO O NO
9	PARA NAND, NANDI2C, Y REINICIO RÁPIDO EXTERNO	NAND Y NANDI2C: NAND ECC REINICIO RÁPIDO EXTERNO DEBE SER 0
8	XIP	ANCHO DE BANDA
7 AL 6	EMAC	MODO FÍSICO
5	APLICA PARA TODOS LOS TIPOS DE BOOT	CLKOUT1 SALIDA ACTIVADA/DESACTIVADA EN XDMA_EVENT_INTRT0
4 AL 0	SECUENCIA DE REINICIO	DEPENDE DE LA SELECCIÓN DEL DISEÑADOR

*Cuadro 11. configuración de pines SYSTEMBOOT.*

*Fuente: Texas Instruments, 2009 [52].*

**El CLKOUT1** es una señal de reloj que es una réplica de la entrada OSC0 que es el oscilador o cristal externo, este no debe de ser usado como salidas de reloj síncronas para ninguna de las interfaces [29]. No se recomienda el uso de este reloj para señales de tiempo críticas en circuitos externos [52]. En el caso del Nodo Multiplataforma no se utiliza esta señal de reloj por lo que se reinicia el sistema desactivada.

El sistema de reinicio del procesador cuenta con un seleccionador de secuencia de reinicio a configurar con los pines del 4 al 0, de acuerdo con el cuadro 12 de configuración de pines **SYSTEMBOOT** la configuración conveniente es:

SYSTEMBOOT PINS	CONFIGURACIÓN
15 AL 14	11B
13 AL 12	00B
11 AL 10	NO IMPORTA PARA ESTA CONFIGURACIÓN
9	NO IMPORTA PARA ESTA CONFIGURACIÓN
8	NO IMPORTA PARA ESTA CONFIGURACIÓN
7 AL 6	NO IMPORTA PARA ESTA CONFIGURACIÓN
5	0B
4 AL 0	11100B

*Cuadro 12. Configuración de los pines SYSTEMBOOT*

*Fuente: Texas Instruments, 2009[52].*

La secuencia de boot se aplica en el orden respectivo que marca la configuración que es MMC1, MMC0, UART0 y USB0; En el caso de que alguna de estas fuentes falle, el bootloader se mueve al siguiente dispositivo en la lista, se debe tomar en cuenta que algunas fuentes de boot toman tiempo antes de cambiar a la siguiente fuente disponible [52].

El bootloader soporta reiniciar desde memorias MMC/SD con las condiciones siguientes [52]:

- a) Memorias MMC/SD que cumplan con la especificación de “Multimedia Card System and Secure I/O” de bajas y altas capacidades.

- b) Memorias MMC/SD deben ser conectadas a MMC0 y MMC1.
- c) Soportar 3.3/1.8V en MMC0 y MMC1.
- d) Iniciar en modo MMC de 1-bit, 4-bit opcional si es que el dispositivo lo soporta.
- e) Frecuencia de reloj de modo identificación a 400 KHz, transferencia de datos de hasta 10 MHz.
- f) Solo una memoria conectada al bus.
- g) Lectura de datos de imágenes directamente de sectores de área de usuario.
- h) Modo de archivo de sistema “File System mode (FAT/12/16/32 soportado con o sin reinicio maestro o Master Boot Record)”, la imagen es leída de un archivo de boot o bootloader.

Puntos importantes a considerar para el boot del procesador[52]:

- a) Es recomendable incluir en la tarjeta prototipo opciones de hardware para poder cambiar los modos de boot para ayudar al desarrollo del sistema. Un ejemplo para considerar es el de la tarjeta BeagleBone Black, la cual implementa un arreglo de resistencias junto con un botón para controlar el modo de reinicio.

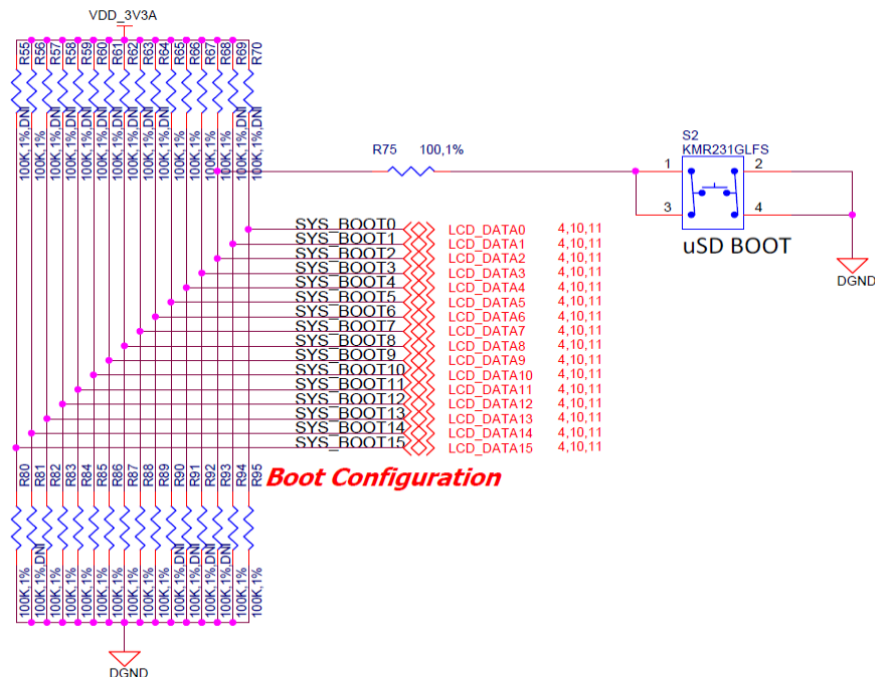


Figura 8. Esquemático de configuración de reinicio de la tarjeta BeagleBone Black.

Fuente: P. No, B. Black y C. Page, 2014[53].

En el diseño de la plataforma BeagleBone Black se utiliza por hardware la siguiente configuración.

<b>SYSBOOT PINS</b>	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
<b>CONFIGURACIÓN</b>	0	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0
<b>DESCRIPCIÓN</b>	24 MHZ		N/A		N/A		N/A	N/A	N/A		HABILITADO	SECUENCIA MMC1, MMC0, UART0, USB0				

Cuadro 13. Configuración de boot de la tarjeta BeagleBone Black.

Fuente: P. No, B. Black y C. Page, 2014[53].

- b) Los pines de boot tienen otras funciones después del reset del procesador, por lo que se debe considerar en el diseño antes de agregar resistencias pullup/down.

### 3.2.2 Selección de Memoria ROM.

De acuerdo con el requerimiento RD-10 y RD-11 el sistema debe tener una memoria ROM externa de 16GB y una ranura para insertar una memoria de tecnología SD.

Si recordamos el modo de boot, el ciclo de carga del sistema operativo comienza por la memoria MMC1 seguido de la MMC0. El procesador cuenta con tres instancias de memoria que son la tarjeta multimedia, o “Multimedia card” por sus siglas en inglés (MMC), Secure Digital (SD), Secure Digital I/O (SDIO) y Módulo de interfaz de alta velocidad o “high speed interfaz module” por sus siglas en inglés (MMCHS).

Estas instancias cuentan con un controlador que provee una interfaz a MMC, tarjeta de memoria SD o SDIO, donde la interfaz de aplicación es responsable de manejar las semánticas de transacción; es decir, para MMC/SDIO se encarga del protocolo a un nivel de transmisión, empaquetado de datos, agregado de CRC, bit de comienzo y término, y revisar la corrección sintética. Este controlador solamente puede soportar una memoria MMC, una SD y una SDIO a la vez [52].

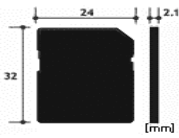
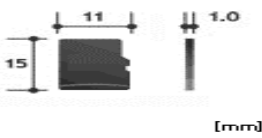
Dicho esto, se determina que solo se puede conectar una memoria por tipo al procesador, es decir solo podemos conectar una memoria de cada una.

### 3.2.2.1 Sistema de interconexión.

Cada interfaz tiene restricciones de reinicio dependiendo de la memoria soportada[52].

- a) MMC0 soporta reinicio desde ranuras de memorias MMC/SD y también desde eMMC/eSD/memorias NAND controladas con menos de 4GB de capacidad. Esto quiere decir que el periférico MMC0 no puede cumplir con el requerimiento RD-10, ya que este pide una capacidad de 16GB y este puerto solo cumple hasta 4GB. Por lo que se determina que este puerto será la interfaz de ranura de memoria SD.

El estándar SD es utilizado en múltiples mercados para el almacenamiento de información en muchos dispositivos electrónicos móviles y fijos [54].

El cuadro 14 muestra las características del estándar SD: FACTOR DE FORMA		SD	MICROSD
DIMENSIÓN			
CAPACIDAD Y TIPO		SD, SDHC, SDXC, Y SDUC	
CAPA FÍSICA	NÚMERO DE PINES	9 PINES	8 PINES
	VOLTAJE DE OPERACIÓN	3.3V VDD RANGO DE LA PRIMERA LÍNEA: 2.7V - 3.6V 1.8V VDD RANGO DE LA SEGUNDA LÍNEA 1.7V-1.95V	
	PROTECCIÓN DE ESCRITURA	SI	NO

Cuadro 14. Descripción de tecnologías SD.

Fuente: SD Standard Overview – SD Association[54].

Las tarjetas convencionales SD tienen solo una línea de 9 pines incluyendo uno a 3.3v de fuente de poder y dos de VSS. Por otro lado, las tarjetas microSD cuentan solo con 8 pines (uno de VSS). Por facilidad de uso por tamaño, se opta por la selección de un conector microSD. La búsqueda del conector se realizó en DigiKey y los criterios de selección son tarjeta tipo

microSD, cantidad de posiciones 8, tipo conector y expulsor. Con base a esto se seleccionó el conector más económico con una cantidad mínima de compra de 1 pieza y con disponibilidad inmediata.

COMPONENTE	DESCRIPCION	COSTO	PROVEEDOR
WM6698CT-ND	MICROSD CARD CONNECTOR	\$ 1.16	DIGIKEY

*Cuadro 15. Selección de conector microSD*

*Fuente: Digikey Electronics, 2019<sup>8</sup>.*

Siendo una memoria microSD externa podemos utilizar capacidades altas de almacenamiento que le permitan al usuario contener el sistema operativo y programas de uso del sistema.

- b) MMC1 soporta reinicio desde eMMC/eSD/memorias NAND controladas con una capacidad de 4GB o mayor. Es decir que el periférico MMC1 será el que contenga el control de la memoria ROM con una capacidad de 16GB.

Debido a que el controlador de memorias del procesador AM3358 solo puede soportar configuraciones donde solamente pueden ser conectadas al sistema un tipo de memoria a la vez, se opta por la selección de la tecnología eMMC. La cual, es un sistema de memoria no volátil integrado, compuesto tanto por memoria flash y su controlador, lo que simplifica el diseño de la interfaz de aplicaciones y libera al procesador host de la gestión de memoria flash a bajo nivel. Para los desarrolladores la eMMC simplifica el proceso de diseño y codificación de la interfaz, lo que resulta en una reducción de tiempo de desarrollo y facilita la compatibilidad con futuras ofertas de dispositivos.

Los criterios de selección del componente van de acuerdo con las restricciones de boot vistas anteriormente.

---

<sup>8</sup> <https://www.digikey.com.mx/product-detail/es/molex/0472192001/WM6698CT-ND/3044147>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

COMPONENTE	DESCRIPCION	COSTO	PROVEEDOR
757-THGAMRG7T13BAIL	EMMC 16GB EMMC 5.1	\$ 14.91	MOUSER

Cuadro 16. Selección de memoria eMMC

Fuente: Mouser Electronics, 2019<sup>9</sup>.

### 3.2.3 Selección de Memoria RAM.

De acuerdo con el requerimiento RD-09 el sistema debe contar con una memoria RAM externa con al menos 2GB de capacidad. El procesador tiene la capacidad de interfaz de los estándares mDDR, DDR2, DDR3, y DDR3L. Las memorias DDR son chips CMOS de alta velocidad que usa una arquitectura de doble procesamiento para lograr transferencias a altas velocidades. Esta arquitectura es esencialmente diseñada con una interfaz para transferir dos palabras de datos por ciclo de reloj en los pines de entradas y salidas [55].

En el actual trabajo seleccionaremos una memoria DDR3L debido a que es la más actual soportada por el procesador y tiene un consumo de energía menor a la DDR3.

COMPONENTE	DESCRIPCION	COSTO	PROVEEDOR
AS4C256M16D3LB-12BCN	DDR3L	\$ 9.79	DIGIKEY

Cuadro 17. Selección de memoria DDR3L

Fuente: Digikey Electronics, 2019<sup>10</sup>.

### 3.2.4 Selección de Módulo WiFi y Bluetooth.

En el mercado podemos encontrar transceiver o módulos de transceptor RF que contienen los protocolos WiFi y Bluetooth en uno mismo. Analizando la tarjeta

<sup>9</sup>

<https://www.mouser.mx/ProductDetail/Kioxia-America/THGAMRG7T13BAIL?qs=sGAEpiMZZMtrOYft0mhKeUzd88m4IAQxZBDVmdZCRPasOz%252BYQcuZ5w%3D%3D>, Mouser Electronics, Inc., índice de productos, diciembre 15 de 2019

<sup>10</sup> <https://www.digikey.com.mx/product-detail/es/alliance-memory-inc/AS4C256M16D3LB-12BCN/1450-1395-ND/6569984>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

BeagleBone Black Wireless esta ya cuenta con un módulo integrado y probado, por lo cual será esta la selección [56].

La selección se realizó de acuerdo con costo y disponibilidad a una pieza<sup>11</sup>.

COMPONENTE	DESCRIPCION	COSTO EN USD	PROVEEDOR
WL1835MODGBMOCT	BLUETOOTH, WIFI 802.11B/G/N, BLUETOOTH 4.2 TRANSCIVER MODULE 2.4GHZ MONTAGE EN SUPERFICIES	\$17.16	DIGIKEY

*Cuadro 18. Selección de modulo Wifi/Bluetooth.*

*Fuente: B. Mpn.[56]*

Este módulo soporta comunicación SDIO de 4-bits que es soportada por el AM3358.

### 3.2.5 Selección de Módulo LoRa.

Como anteriormente revisamos LoRa es un desarrollo de hardware de comunicación RF de largo alcance, y además cabe destacar que cuenta con un protocolo de bajo consumo de energía sin mermar su eficiencia.

Las características de búsqueda son de un módulo de rápida aplicación que nos permita validar el protocolo montado sobre el Nodo Multiplataforma, además del costo bajo y su disponibilidad desde una pieza.

COMPONENTE	DESCRIPCION	COSTO EN USD	PROVEEDOR
RN2483A-I/RM104-ND	MODULO LORA	\$ 12.80	DIGIKEY

*Cuadro 19. de selección de modulo LoRa.*

*Fuente: Elaboración propia.*

### 3.2.6 Selección de conectores USB.

El controlador USB del procesador provee un mecanismo de transferencia para dispositivos USB con una velocidad de hasta 480 Mbps. El subsistema de USB

---

<sup>11</sup> <https://www.digikey.com.mx/products/es?keywords=WL1835MODGBMOCT>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.



contiene dos USB 2.0 independientes acoplados a dos controladores OTG, los dos controladores soportan ser host o periférico para que la configuración sea a preferencia del usuario [52]. En el presente trabajo se utilizarán los dos dispositivos USB configurando uno como host para manipulación del sistema operativo y el otro será periférico para poder agregar dispositivos de control del Nodo.

COMPONENTE	DESCRIPCION	COSTO EN USD	PROVEEDOR
87520-0010BLF	USB HOST	\$ 0.73	DIGIKEY
UX60SC-MB-5ST (80)	USB MINI B	\$ 0.80	DIGIKEY

Cuadro 20. Selección de conectores USB

Fuente: Digikey Electronics, 2019<sup>12,13</sup>.

### 3.2.7 Multiplexado de pines.

El procesador AM335x contiene varias interfaces de comunicación, y para reducir costos y mantener su máxima funcionalidad, aproximadamente la mitad de las terminales del procesador se pueden multiplexar con hasta ocho señales diferentes, en específico, existen varias combinaciones de multiplexado en algunos grupos llamados “IO sets”.

Para la selección de los puertos sobre el procesador, Texas instruments tiene una herramienta llamada “Pin Mux Utility” que ayuda al diseñador del sistema a seleccionar adecuadamente la configuración de pines [51]. Este se puede utilizar en línea de la página de Texas instruments<sup>14</sup>.

COMPONENTE	DESCRIPCION	COSTO EN USD	PROVEEDOR
AM3358BZCZ100	PROCESADOR	\$ 17.21	DIGIKEY

Cuadro 21. Costo de Procesador AM3358.

Fuente: Elaboración propia.

<sup>12</sup> [https://www.digikey.com.mx/products/es?keywords=UX60SC-MB-5ST\(80\)](https://www.digikey.com.mx/products/es?keywords=UX60SC-MB-5ST(80)), Digi-Key electronics, índice de productos, diciembre 15 de 2019.

<sup>13</sup> <https://www.digikey.com.mx/products/es?keywords=87520-0010BLF>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

<sup>14</sup> <https://dev.ti.com/pinmux/app.html#/default>, Texas Instruments, pinmux, diciembre 15 de 2019

La herramienta solicita características específicas del procesador, es necesario seleccionar un número de parte como se muestra en la Figura 9, en nuestro caso es el descrito en el cuadro 21.

Welcome! To get started with TI PinMux, please use one of the below options to start or continue your design.

▼ Start a new Design

Board:	N/A	▼
Device:	AM335x	▼
Part:	Default	▼
Package:	ZCZ	▼

**Start**

Figura 9. Configuración de procesador a multiplexar.

Fuente: Texas Instruments, Pinmux, 2019<sup>15</sup>.

Lo siguiente es presionar start para que nos dirija a la página de configuración de multiplexado como se muestra en la Figura 10 Configuración de multiplexado.

---

<sup>15</sup> [https://dev.ti.com/pinmux/app.html#/config/undefined/AM335x/ID\\_500/ID\\_0/null/null](https://dev.ti.com/pinmux/app.html#/config/undefined/AM335x/ID_500/ID_0/null/null), Texas Instruments, pinmux, diciembre 15 de 2019

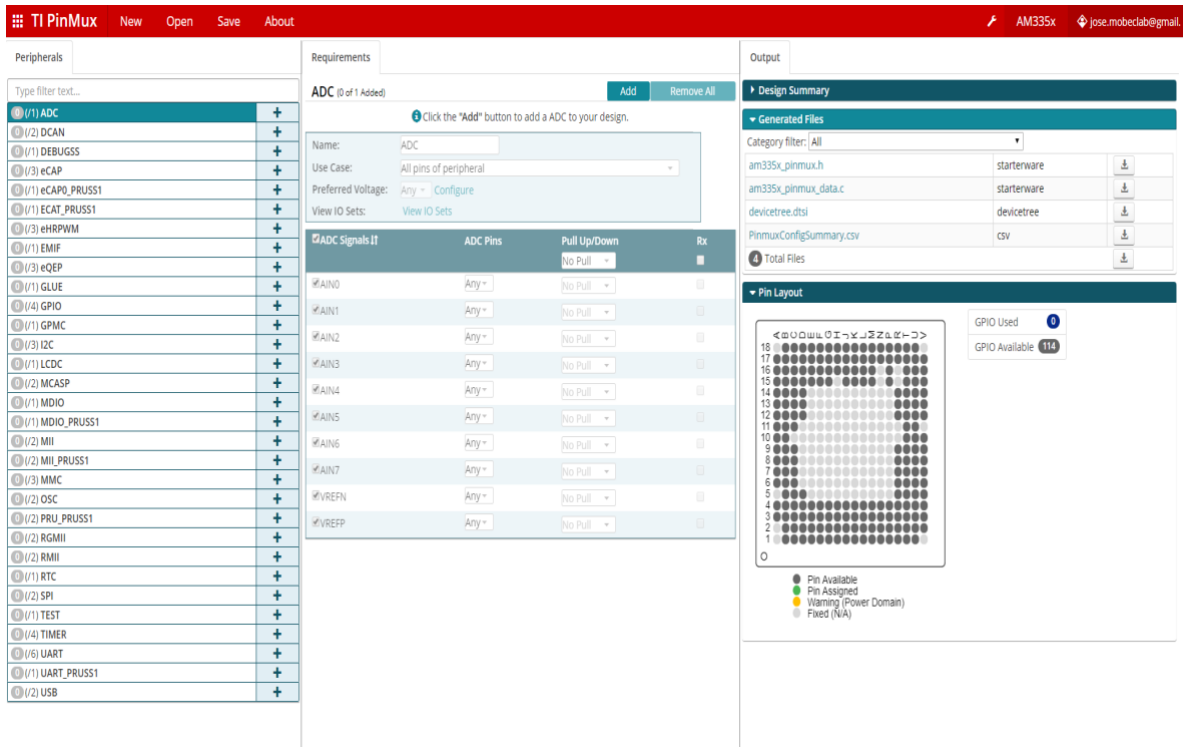


Figura 10. Configuración de multiplexado

Fuente: Texas Instruments, Pinmux, 2019<sup>16</sup>.

En la Figura 11 comenzamos a seleccionar los módulos de hardware que utilizaremos y las configuraciones de pines más convenientes para el desarrollo del sistema.

<sup>16</sup> [https://dev.ti.com/pinmux/app.html#/config/undefined/AM335x/ID\\_500/ID\\_0/null/null](https://dev.ti.com/pinmux/app.html#/config/undefined/AM335x/ID_500/ID_0/null/null), Texas Instruments, pinmux, diciembre 15 de 2019

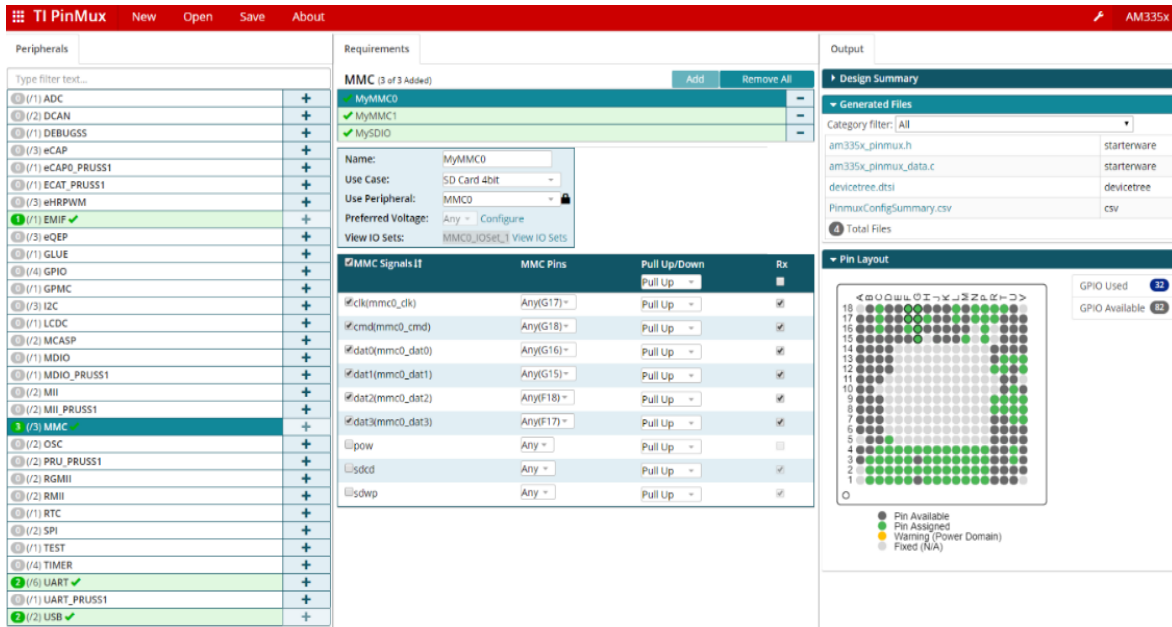


Figura 11. Configuración de pines del procesador.

Fuente: Texas Instruments, Pinmux, 2019<sup>17</sup>.

Como se puede observar en la Figura 11 Configuración de pines del procesador, la herramienta realiza la distribución de pines de acuerdo con los módulos que elijas, y en caso de que se tenga algún conflicto te hace saber en qué módulo y pin existe.

Una vez terminando la configuración de pines se puede proceder al diseño electrónico y de programación, para cada zona configurada del procesador como se muestra en el siguiente capítulo.

<sup>17</sup> [https://dev.ti.com/pinmux/app.html#/config/undefined/AM335x/ID\\_500/ID\\_0/null/null](https://dev.ti.com/pinmux/app.html#/config/undefined/AM335x/ID_500/ID_0/null/null), Texas Instruments, pinmux, diciembre 15 de 2019



## Capítulo 4

# Implementación y resultados

## Capítulo 4. Implementación y resultados

La implementación y los resultados se basan en un sistema operativo integrado por el fabricante de la tarjeta BeagleBone Black.

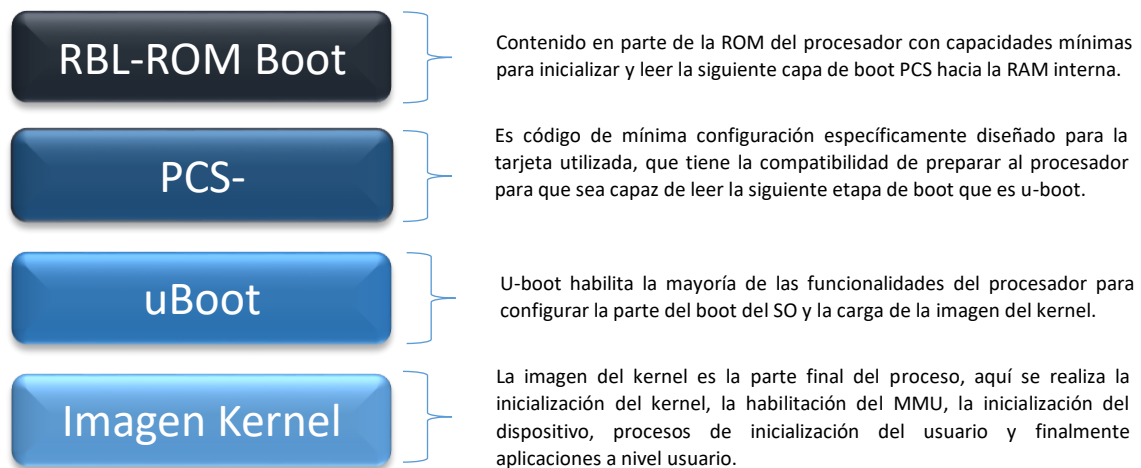
### 4.1 Proceso de inicio del sistema operativo embebido basado en linux.

La mayoría de los procesadores reinician a través de un método de carga en cadena donde cada capa le da mayor funcionalidad al proceso. Como ya lo revisamos en el capítulo 2 la memoria ROM interna cuenta con un bootloader que tiene la capacidad de cargar la siguiente etapa de reinicio.

El proceso de carga del bootloader del AM3358 es como sigue [52]:

1. Se crea una lista de dispositivos de boot que consiste en todos los dispositivos que serán buscados y accesados en localización de una imagen de reinicio. Esta lista se llena con base a la configuración realizada sobre el SYSBOOT.
2. Ya que se realice la lista de dispositivos, la rutina de búsqueda del bootloader examina los dispositivos enumerada secuencialmente, ya sea un boot por memoria o por periférico. El boot de memoria es ejecutado cuando el dispositivo es de tipo NOR, NAND, MMC o SPI-EEPROM, estas son las opciones de direccionamiento que ofrece el procesador. Por otra parte, el boot por medio de periférico es ejecutado cuando el dispositivo es de tipo ethernet, USB o UART. En el caso del nodo multiplataforma solo se requiere ejecutar el modo de boot por memoria.
3. El procedimiento del boot por memoria lee información de un dispositivo tipo memoria. Si una imagen de boot válida es encontrada y leída exitosamente, el código comienza a ejecutarse.
4. En caso de que todos los dispositivos de la lista fallen, el bootloader entra en un loop esperando a que el watchdog reinicie el sistema.

El bootlaoder precargado en ROM no conoce el sistema que se cargará en memoria, por lo que utiliza métodos predefinidos para encontrar el siguiente código, capa, o etapa de boot. La memoria ROM y RAM interna son limitadas en tamaño y es por eso que el proceso de boot es separado en tres etapas para poder realizar la carga del sistema operativo en la tarjeta [57]<sup>18</sup>.



Cuadro 22. Proceso de reinicio del procesador AM3358.

Fuente: *Giving Linux the Boot*, 2012[57].

El presente trabajo valida la integración de hardware con base a la plataforma BeagleBone Black, el fabricante provee el sistema operativo Debian en diferentes versiones acoplado previamente a la tarjeta y ya convertido en imagen, por lo que en el caso de la validación de la carga de sistema operativo se realizará de acuerdo con las indicaciones del fabricante de la plataforma, la cual se basa en los procesos de carga mencionados en el cuadro 22<sup>19</sup>.

Para validación de hardware del Nodo Multiplataforma una vez que sea manufacturado se tiene que crear una distribución del sistema operativo que se

<sup>18</sup> [https://training.ti.com/system/files/docs/sitara\\_boot\\_camp\\_03\\_giving\\_linux\\_the\\_boot.pptx](https://training.ti.com/system/files/docs/sitara_boot_camp_03_giving_linux_the_boot.pptx), Sitara Boot Camp, Texas Instruments, diciembre 15 de 2019

<sup>19</sup> <https://beagleboard.org/getting-started#update>, BeagleBoard.org, getting-started, diciembre 15 de 2019

acople al hardware de la tarjeta, en este caso se recomienda utilizar Yocto project<sup>20</sup>. Uno de los alcances de este proyecto es el de validar el hardware por medio de la plataforma BeagleBone Black, en la cual el fabricante ya incluye un sistema operativo preinstalado de fábrica en la memoria eMMC, en el caso de querer realizar una actualización el usuario puede seguir los pasos recomendados por el fabricante<sup>21</sup>. La plataforma ya cuenta con los drivers de WiFi, Bluetooth, SD y USB preinstalados por lo que solo resta realizar la configuración y validar la comunicación.

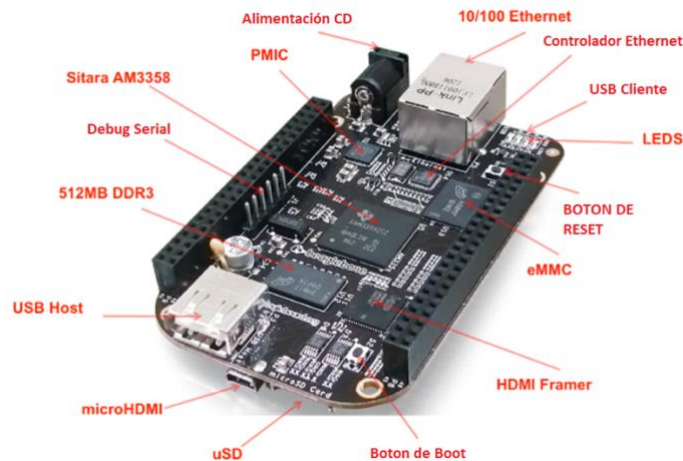


Figura 12. Detalles de Hardware de BeagleBone Black

Fuente: BeagleBoard.org, support, bone101<sup>22</sup>.

## 4.2 Pruebas de comunicación

Para corroborar la correcta selección y funcionamiento de componentes se realiza una prueba de comunicación. Es importante mencionar que la tarjeta ya contiene

---

<sup>20</sup> <https://www.yoctoproject.org/>, Linux Foundation and Yocto Project, diciembre 15 de 2019.

<sup>21</sup> <https://beagleboard.org/Support/BoneScript/updates/>, BeagleBoard.org, Software Updates, diciembre 15 de 2019.

<sup>22</sup> <https://beagleboard.org/Support/bone101>, BeagleBoard.org, bone101, diciembre 15 de 2019.



un sistema operativo instalado de fábrica con los controladores de los periféricos preinstalados.

## 4.2.1 Comunicación USB

Para validar la comunicación vía USB basta con conectar la tarjeta al conector USB a la PC y utilizar el programa putty para realizar una conexión al dispositivo vía SSH. Este es un protocolo de comunicación que enlaza de manera segura a dos dispositivos utilizando una arquitectura cliente servidor que permite al usuario conectarse a un host de manera remota<sup>23</sup>.

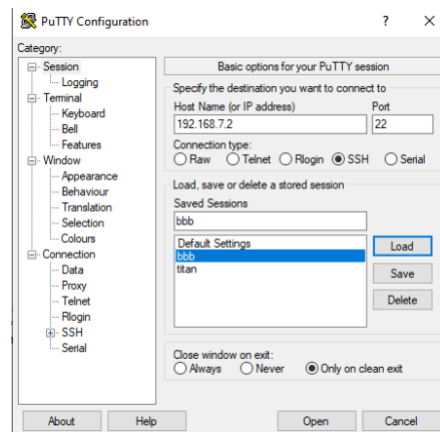


Figura 13. Configuración putty para conexión SSH de la BeagleBone Black.

Fuente: Programa Putty, elaboración propia.

Se presiona open para iniciar sesión.

23

[https://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ssh.html#:~:text=SSH%E2%84%A2%20\(o%20Secure%20SHell,conectarse%20a%20un%20host%20remotamente., Red Hat, Inc., Protocolo SSH, junio 30 de 2020.](https://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ssh.html#:~:text=SSH%E2%84%A2%20(o%20Secure%20SHell,conectarse%20a%20un%20host%20remotamente., Red Hat, Inc., Protocolo SSH, junio 30 de 2020.)

```

192.168.7.2 - PuTTY
login as: mobec
mobec@192.168.7.2's password:
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Dec 1 20:46:30 2019 from 192.168.7.1
mobec@beaglebone:~$

```

Figura 14. Inicio de sesión con putty.

Fuente: Programa Putty, elaboración propia.

De esta manera validamos la comunicación USB cliente, el cual nos permite tener acceso al sistema operativo, el protocolo USB nos permite conectar a la tarjeta extensiones o HUB para poder conectar más dispositivos a la misma, pueden ser desde un teclado, ratón u dispositivo a preferencia de usuario o necesidad del sistema.

## 4.2.2 Comunicación WiFi

La tarjeta BeagleBone Black Wireless cuenta con el mismo tranciever seleccionado para la tarjeta Nodo Multiplataforma de Conectividad para IoT. Para realizar la configuración se utiliza una interfaz de comandos “Connmanctl”, como se muestra en Figura 15, este puede manejar la mayoría de las conexiones y puede ser usado para habilitar y deshabilitar diferentes tecnologías que existen en el sistema. En nuestro caso lo utilizamos para habilitar la red de WiFi en la tarjeta. En el Cuadro 23 se describe el uso de los comandos de configuración del modulo WiFi.

COMANDO	DESCRIPCION
connmanctl	Abre línea de comandos para configurar el módulo WiFi
scan wifi	Busca redes WiFi-disponibles
enable wifi	Habilita la red WiFi
services	Despliega las redes WiFi -disponibles
agent on	Utilizado para habilitar el ingreso de la red WiFi y su contraseña
connect	Conecta a la red WiFi -especificada
quit	Termina la sesión en la línea de comando

Cuadro 23. Descripción de comandos para la configuración de módulo WiFi.

Fuente: Elaboración propia.

Para comenzar con la configuración, primero entramos a la línea de comandos ingresando “Connmanctl”, enseguida llamaremos a los servicios disponibles de conexión, el cual nos proporciona una lista de la cual podemos seleccionar al que nos conectaremos, en nuestro caso es “INFINITUM1234\_2.4”, ahora habilitamos el agente para poder proceder con la conexión, en nuestro caso el sistema ya tenía guardada una contraseña, pero en el caso de no tenerla habrá que introducirla.

```
root@beaglebone:~# connmanctl
Error getting VPN connections: The name net.connman.vpn was not provided by any .service filescon
connmanctl> enable wifi
Error wifi: Already enabled
connmanctl> scan wifi
Scan completed for wifi
connmanctl> services
*A INFINITUM1234 2.4      wifi_8030dcfd7d0a_494e46494e4954554d313233345f322e34_managed_psk
  DIRECT-Uc-BRAVIA     wifi_8030dcfd7d0a_4449524543542d55632d425241564941_managed_psk
  INFINITUM35szs       wifi_8030dcfd7d0a_494e46494e4954554d33357a73_managed_psk
  INFINITUMtaks        wifi_8030dcfd7d0a_494e46494e4954554d74616b73_managed_psk
  ARRIS-7142           wifi_8030dcfd7d0a_41525249532d37313432_managed_psk
  INFINITUME47A2F      wifi_8030dcfd7d0a_494e46494e4954554d453437413246_managed_psk
connmanctl> agent on
Agent registered
connmanctl> connect wifi_8030dcfd7d0a_494e46494e4954554d313233345f322e34_managed_psk
Connected wifi_8030dcfd7d0a_494e46494e4954554d313233345f322e34_managed_psk
connmanctl>
```

Figura 15. Conexión WiFi por BeagleBone Black

Fuente: Programa Putty, elaboración propia<sup>24</sup>.

Para la prueba de acceso a internet actualizaremos el sistema operativo por medio del siguiente comando “sudo apt-get update”, como se muestra en la Figura 16, el cual nos arroja el progreso de la actualización de manera exitosa si es que ya se tiene acceso a internet como se muestra en la Figura 16 Pruebas de conexión WiFi.

---

<sup>24</sup> <https://dougreesse.com/beaglebone-black-wireless-initial-setup-and-wifi-connect.html>, [Doug Reese](#), BeagleBone Black Wireless – Initial setup and wifi connect, diciembre 15 de 2019.

```

root@beaglebone:~# connmanctl
Error getting VPN connections: The name net.connman.vpn was not provided by any .service filescon
connmanctl> enable wifi
Error wifi: Already enabled
connmanctl> scan wifi
Scan completed for wifi
connmanctl> services
*A INFINITUM1234_2.4 wifi_8030dcfd7d0a_494e46494e4954554d313233345f322e34_managed_psk
  DIRECT-Uc-BRAVIA wifi_8030dcfd7d0a_4449524543542d55632d425241564941_managed_psk
  INFINITUM35zs wifi_8030dcfd7d0a_494e46494e4954554d33357a73_managed_psk
  INFINITUMtaks wifi_8030dcfd7d0a_494e46494e4954554d74616b73_managed_psk
  ARRIS-7142 wifi_8030dcfd7d0a_41525249532d37313432_managed_psk
  INFINITUME47A2F wifi_8030dcfd7d0a_494e46494e4954554d453437413246_managed_psk
connmanctl> agent on
Agent registered
connmanctl> connect wifi_8030dcfd7d0a_494e46494e4954554d313233345f322e34_managed_psk
Connected wifi_8030dcfd7d0a_494e46494e4954554d313233345f322e34_managed_psk
connmanctl> quit
root@beaglebone:~# sudo apt-get update
Ign:1 http://deb.debian.org/debian stretch InRelease
Get:2 http://deb.debian.org/debian stretch-updates InRelease [91.0 kB]
Get:3 http://repos.rcn-ee.com/debian stretch InRelease [3,064 B]
Get:4 http://deb.debian.org/debian-security stretch/updates InRelease [94.3 kB]
Get:5 http://repos.rcn-ee.com/debian stretch/main armhf Packages [1,109 kB]
Hit:6 http://deb.debian.org/debian stretch Release
Get:7 http://deb.debian.org/debian stretch-updates/main armhf Packages.diff/Index [13.0 kB]
Get:8 http://deb.debian.org/debian stretch-updates/main armhf Packages 2019-11-06-2017.59.pdiff [
903 B]
Get:8 http://deb.debian.org/debian stretch-updates/main armhf Packages 2019-11-06-2017.59.pdiff [
903 B]
Get:9 http://deb.debian.org/debian-security stretch/updates/main armhf Packages [486 kB]
Fetched 1,797 kB in 36s (49.8 kB/s)
Reading package lists... Done
root@beaglebone:~# ping www.google.com
PING www.google.com(2001:4860:4802:32::75 (2001:4860:4802:32::75)) 56 data bytes
64 bytes from 2001:4860:4802:32::75 (2001:4860:4802:32::75): icmp_seq=1 ttl=57 time=272 ms
64 bytes from 2001:4860:4802:32::75 (2001:4860:4802:32::75): icmp_seq=2 ttl=57 time=192 ms
64 bytes from 2001:4860:4802:32::75 (2001:4860:4802:32::75): icmp_seq=3 ttl=57 time=127 ms
64 bytes from 2001:4860:4802:32::75 (2001:4860:4802:32::75): icmp_seq=4 ttl=57 time=44.0 ms

--- www.google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 44.076/159.224/272.959/84.093 ms
root@beaglebone:~# █

```

Figura 16. Pruebas de conexión WiFi.

Fuente: Elaboración propia.

Otra manera de probar el correcto acceso a internet, también podemos introducir el comando “ping”, que como se muestra en la Figura 16, este realiza una verificación del estado de la conexión de la red, entre anfitrión local y remoto. Este comando es repetitivo por lo que se tendrá que detener presionando ctrl + c.

### 4.2.3 Comunicación Bluetooth

Al igual que el controlador de WiFi connmanctl, bluetooth cuenta con bluetoothctl, el cual se encarga de las configuraciones a nivel hardware, el despliegue se muestra en la Figura 17. Utilizando este controlador arrancamos la

línea de comando. Seguido revisamos los dispositivos detectados, se realizó un intento de conexión a un dispositivo erróneo, el cual nos arrojó un dispositivo sin disponibilidad.

COMANDO	DESCRIPCIÓN
bluetoothctl	Abre línea de comandos para configurar el módulo bluetooth
scan on	Detecta los dispositivos disponibles para conexión
connect	Conecta al dispositivo seleccionado

Cuadro 24. Descripción de comandos para la configuración de módulo Bluetooth.

Fuente: Elaboración propia.

```

root@beaglebone:~# bluetoothctl
[NEW] Controller 80:30:DC:FD:7D:09 beaglebone [default]
[bluetooth]# Agent registered
Invalid command
[bluetooth]# scan on
Discovery started
[CHG] Controller 80:30:DC:FD:7D:09 Discovering: yes
[NEW] Device 78:70:56:61:F5:5D 78-70-56-61-F5-5D
[bluetooth]# connect
Missing device address argument
[bluetooth]# connect 80:30:DC:FD:7D:09
Device 80:30:DC:FD:7D:09 not available
[CHG] Device 78:70:56:61:F5:5D ManufacturerData Key: 0x004c
[CHG] Device 78:70:56:61:F5:5D ManufacturerData Value: 0x10
[CHG] Device 78:70:56:61:F5:5D ManufacturerData Value: 0x05
[CHG] Device 78:70:56:61:F5:5D ManufacturerData Value: 0x4b
[CHG] Device 78:70:56:61:F5:5D ManufacturerData Value: 0x1c
[CHG] Device 78:70:56:61:F5:5D ManufacturerData Value: 0x47
[CHG] Device 78:70:56:61:F5:5D ManufacturerData Value: 0x24
[CHG] Device 78:70:56:61:F5:5D ManufacturerData Value: 0xb0
[NEW] Device F0:99:B6:1B:DB:57 iPepos

```

Figura 17. Configuración de controlador Bluetooth.

Fuente: Elaboración propia.

El controlador sigue mostrando los dispositivos en alcance, el de interés es iPepos por lo que intentamos la conexión, pero fue rechazada, como se muestra en la Figura 18.

```
[NEW] Device F0:99:B6:1B:DB:57 iPepos
[bluetooth]# connect F0:99:B6:1B:DB:57
Attempting to connect to F0:99:B6:1B:DB:57
Failed to connect: org.bluez.Error.Failed
```

Figura 18. Falla de conexión Bluetooth.

Fuente: Elaboración propia.

En la Figura 19 se muestra nuevamente el intento de conexión, esta vez resultando en correcta y aceptada, el dispositivo al que se conectó es un iPhone. De esta manera validamos la correcta comunicación del tranciever bluetooth.

```
[bluetooth]# connect F0:99:B6:1B:DB:57
Attempting to connect to F0:99:B6:1B:DB:57
[CHG] Device F0:99:B6:1B:DB:57 Connected: yes
[CHG] Device F0:99:B6:1B:DB:57 Modalias: bluetooth:v004Cp7230d0D10
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 00000000-deca-fade-deca-deafdecacafe
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 00001000-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 0000110a-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 00001116-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 0000111f-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 0000112f-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 00001132-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 00001200-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 00001801-0000-1000-8000-00805f9b34fb
[CHG] Device F0:99:B6:1B:DB:57 UUIDs: 02030302-1d19-415f-86f2-22a2106a0a77
[CHG] Device F0:99:B6:1B:DB:57 ServicesResolved: yes
```

Figura 19. Conexión exitosa bluetooth.

Fuente: Elaboración propia.

#### 4.2.4 Diseño de tarjeta LoRa

La tarjeta BeagleBone Black no cuenta con tecnología LoRa integrada por lo que se opta por diseñar una tarjeta que contiene el tranciever RN2483A de microchip que es una tecnología que provee comandos de interface para un rápido desarrollo, se comunica por medio de UART, sus comandos permiten acceder a la capa física de LoRa y a la capa de LoRaWAN clase A y C [58]. El módulo RN2483A lleva a cabo estas funciones de acuerdo con la siguiente arquitectura.

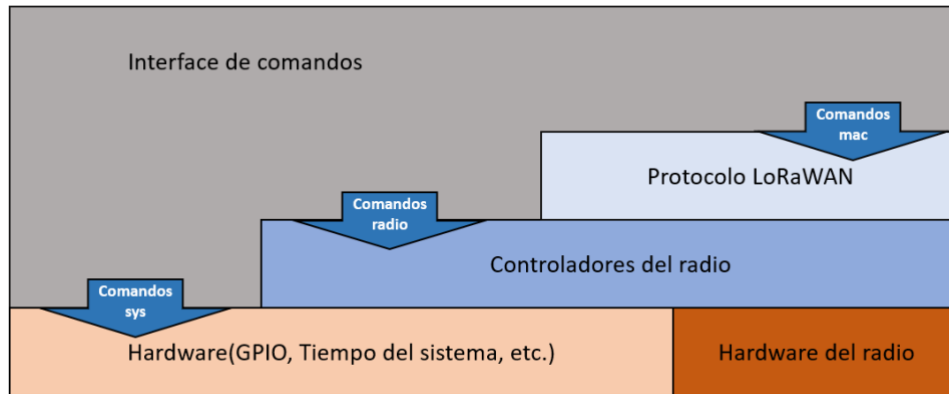


Figura 20. Arquitectura de firmware del RN2483A.

Fuente: Microchip Technology Inc., 2017[58].

Los comandos de esta arquitectura se dividen en comandos mac, radio y sys de acuerdo con la Figura 20. Esto nos permite utilizar las tres diferentes capas de firmware teniendo la versatilidad de solo usar una capa sin involucrar a las demás.

#### 4.2.5 Diseño de hardware

El módulo utiliza un voltaje de entre 2.1v y 3.6v con un consumo de corriente máximo de 38.9mA a un voltaje de 3.3v. Dadas estas condiciones se opta por reservar cuatro pines de entrada de voltaje y tierra en la tarjeta LoRa ya que la tarjeta BeagleBone Black cuenta con salidas de voltaje de 3.3v que utilizaremos como fuente de poder para energizar el módulo, además la mayoría de los dispositivos electrónicos actuales cuentan con entradas y salidas de voltaje a 3.3v.

Para acoplar el voltaje recibido es necesario añadir un regulador de voltaje de DC a DC, en nuestro caso siendo que el voltaje de entrada es muy cercano al de salida es necesario elegir un regulador de baja caída de tensión o mejor conocido como “Low Drop Out Voltage” por sus siglas en inglés LDO que proteja el circuito de cualquier variación de voltaje a la entrada y además que permita que el voltaje de salida este dentro de los rangos permitidos por el RN2483A que son 2.1v a 3.6v absorbiendo la corriente del mismo.

La ventaja de los LDOs es que pueden regular voltajes de entrada a voltajes de salida muy cercanos, en realidad la mayoría de los LDOs utilizan FETs NPN o PNP que nos ayudan a mantener el voltaje deseado con voltajes de caída menores a 100mV [59].

La caída de tensión en el LDO está dada por:

$$V_{\text{decaída}} = V_{\text{entrada}} - V_{\text{salida}}$$

De aquí podemos obtener que los límites de selección del LDO son:

$$V_{\text{decaída}} = 3.3\text{v} - 2.1\text{v} = 1.2\text{v}$$

Es decir que el LDO no puede tener un voltaje de caída mayor a 1.2volts para cumplir con los parámetros del transceiver LoRa. Dado esto se propone el número de parte 296-21569-1-ND con un voltaje de caída de 115mV a un consumo de corriente de 50mA [60], el RN2483A tiene un consumo máximo de 38.9mA a uso de mayor potencia de transmisión, por lo que tenemos 11.1mA sobrados a esta caída de tensión. Dados estos valores:

$$V_{\text{dropout}} = 3.3\text{v} - 0.115\text{v} = 3.185\text{v}$$

Son 3.185v de salida que están dentro del rango de alimentación del RN2483A, por lo que este LDO es una buena opción de acoplamiento de voltaje, además en caso de no utilizar la potencia máxima del transceiver el consumo de corriente tiende a disminuir.

COMPONENTE	DESCRIPCION	COSTO EN USD	PROVEEDOR
296-21569-1-ND	REGULADOR LOW DROP OUT	\$1.13	DIGIKEY

*Cuadro 25. Selección de regulador LDO*

*Fuente: Digi Key Electronics, 2019<sup>25</sup>.*

<sup>25</sup> <https://www.digikey.com.mx/products/es?keywords=296-21569-1-ND>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.



El fabricante propone una implementación del regulador como sigue:

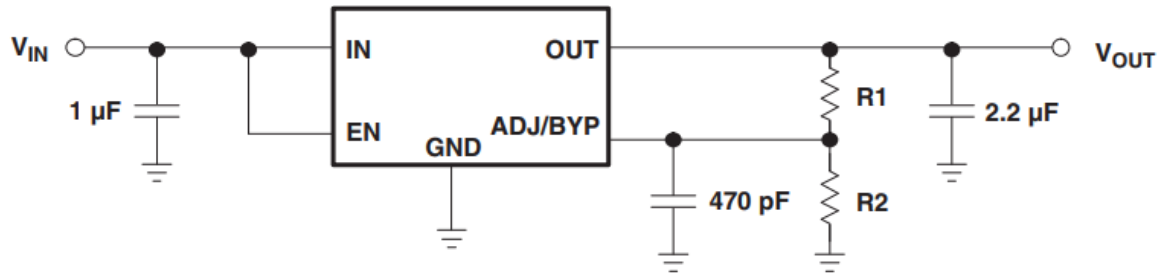


Figura 21. Esquemático propuesto por el fabricante del LDO.

Fuente: A.O. Voltage, 2007[60].

Este regulador con número de parte del fabricante TL5209 tiene un rango de entrada de 2.5v a 16v a una corriente máxima de 500mA, la salida de voltaje puede ser ajustada usando las resistencias R1 y R2 y tiene una exactitud del 1% al 2% dependiendo de la temperatura ambiente, el valor máximo de caída de voltaje es de 10mV a 500mV dependiendo de la corriente de carga de la salida. El diseño de hardware del regulador se basa en las siguientes características [60].

1. Cuando  $EN > 2V$  el regulador está en activo, de lo contrario el regulador se apaga. Si no hay necesidad de habilitar el apagado de protección, EN debe estar conectado a  $V_{IN}$  (voltaje de entrada).
2. Para prevenir una carga rápida de voltaje al inicio de operación es necesario proteger al amplificador de error y al circuito de voltaje de referencia, ya que estos no tienen una rápida respuesta inicial, es necesario darles tiempo para responder. Para evitar una sobre carga inicial el fabricante recomienda utilizar capacitores de  $2.2\ \mu\text{F}$  y  $470\ \text{pF}$  para la salida y la entrada de bypass respectivamente. Esta recomendación de diseño es para aplicaciones sensibles a la salida de voltaje ( $V_{OUT}$ ), en nuestro caso el voltaje de salida se requiere para alimentación y no es crítica en tiempo de respuesta inicial, sin embargo, agregaremos esta recomendación en el diseño de la tarjeta.
3. El capacitor de entrada debe ser de  $1\ \mu\text{F}$ .

4. De acuerdo con la Figura 12. El LDO generan voltaje de referencia de 1.242v entre el voltaje de salida y el pin ADJ, dado esto:

$$V_{OUT} = 1.242V * (1 + R2/R1)$$

Si se propone una resistencia comercial de 470kohm entonces:

$$R1 = 283.132k$$

Ahora podemos buscar una resistencia comercial que se le aproxime que es de 280k.

COMPONENTE	DESCRIPCION	COSTO EN USD	PROVEEDOR
CC0805KKX7R5BB105	CAPACITOR DE 1UF	\$0.26	DIGIKEY
732-7646-1-ND	CAPACITOR DE 2.2UF	\$0.19	DIGIKEY
GRM216R71H471KA01J	CAPACITOR DE 470 PF	\$0.12	DIGIKEY
A126370CT-ND	RESISTENCIA DE 470KOHMS +/-1%	\$0.1	DIGIKEY
311-280KCRCT-ND	RESISTENCIA DE 280KOHMS +/-1%	\$0.1	DIGIKEY

Cuadro 26. Selección de capacitores de desacoplo y resistencias de ajuste de voltaje

Fuente: Digi Key Electronics, 2019<sup>26 27 28 29</sup>.

Es importante destacar que al LDO se le puede alimentar en VIN con más de 3.3v con la consideración que el voltaje de caída será más alto y por ende la disipación de temperatura incrementará, ya que la corriente consumida será mayor, debido a esto se recomienda el cálculo de las resistencias de acuerdo con una alimentación fija.

<sup>26</sup> <https://www.digikey.com.mx/products/es?keywords=CC0805KKX7R5BB105%E2%80%8E>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

<sup>27</sup> <https://www.digikey.com.mx/products/es?keywords=%E2%80%8E732-7646-1-ND>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

<sup>28</sup> <https://www.digikey.com.mx/products/es?keywords=490-11574-1-ND>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

<sup>29</sup> <https://www.digikey.com.mx/products/es?keywords=A126370CT-ND>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

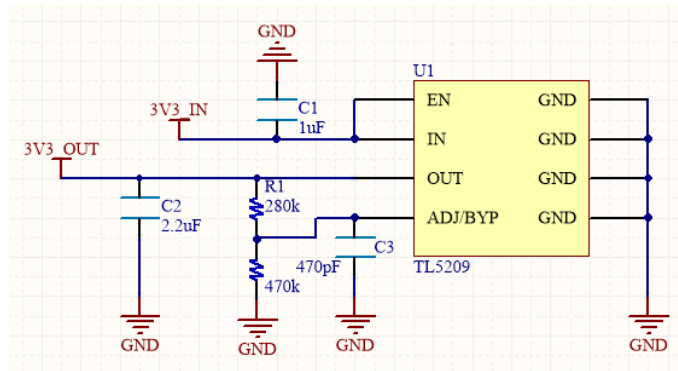


Figura 22. Esquemático del LDO.

Fuente: Elaboración propia.

Por otro lado, el tranciver cuenta con 13 pines de control llamados GPIOs de los cuales 12 se pueden configurar como entradas y salidas digitales o analógicas. En específico el GPIO4 solo se puede utilizar como entrada o salida digital. De aquí se opta por aprovechar esto GPIOs para agregar cuatro LEDs y dos botones “Push” que nos permitan manipular el sistema a conveniencia del usuario.

COMPONENTE	DESCRIPCION	COSTO EN USD	PROVEEDOR
475-1410-1-ND	LED VERDE	\$0.27	DIGIKEY
A129741CT-ND	RESISTENCIAS 220 OHM	\$0.1	DIGIKEY
RR12P10.0KDCT-ND	RESISTENCIAS DE 10K	\$0.11	DIGIKEY
P8047SCT-ND	BOTONES	\$0.38	DIGIKEY

Cuadro 27. Selección de LED, Resistencias y Botones Push

Fuente: Digi Key Electronics, 2019<sup>30 31 32</sup>.

El consumo de corriente de los LEDs con una resistencia de 220 ohm es de 15mA, la tarjeta cuenta con 5 LEDs, cuatro de control y uno de indicador de

<sup>30</sup> <https://www.digikey.com.mx/products/es?keywords=475-1410-1-ND>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

<sup>31</sup> <https://www.digikey.com.mx/products/es?keywords=A129741CT-ND>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

<sup>32</sup> <https://www.digikey.com.mx/products/es?keywords=P8047SCT-ND>, Digi-Key electronics, índice de productos, diciembre 15 de 2019.

encendido, por lo que su consumo es de 75mA extras al consumo del módulo, esto en total nos genera un consumo constante máximo de 113.9 mA. Debido a este consumo, la caída de voltaje incrementará a 150mV, por lo que el VOUT será igual a 3.15v lo cual sigue estando dentro del rango de operación del RN2483A.

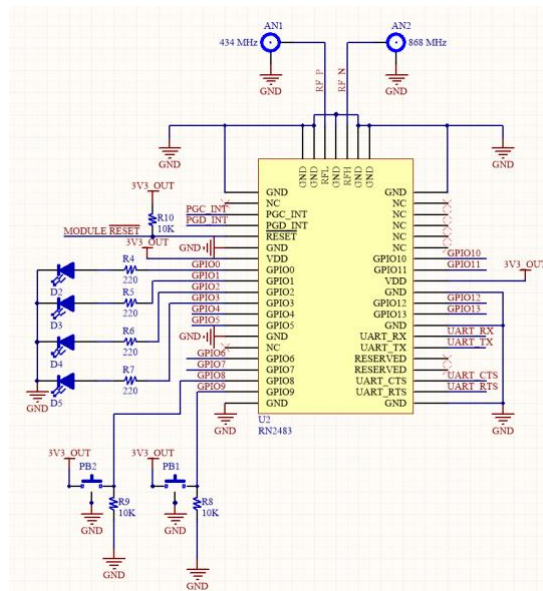


Figura 23. Esquemático del RN2483A.

Fuente: Elaboración propia.

Además, para aprovechar la capacidad de control con la que cuenta el módulo se agregó una tira de pines que nos ayudan a poder interactuar con la tarjeta a preferencia del usuario.

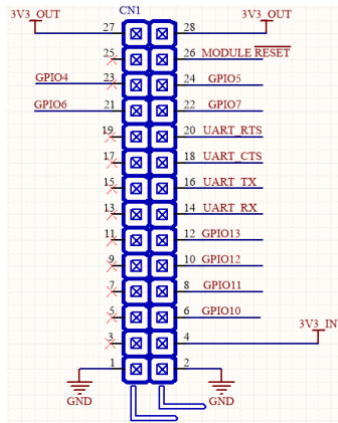


Figura 24. Esquemático de tira de pines de tarjeta LoRa.

Fuente: Elaboración propia.

El transceiver RN2483A opera en el rango de frecuencias de 433MHz y 868 MHz, para el diseño de la tarjeta de circuito impreso el diseño de las antenas se basó en las recomendaciones del fabricante. Este diseño está basado en la conexión de antenas externas por medio de sockets, como se puede observar en la Figura 25 Layout de antenas Lora. El diseño del PCB recomendado para un correcto funcionamiento es como sigue:

### Detalle del PCB

Dos capas

Perforaciones plateadas

Material FR4

Grosor de 1.55mm

Plateado de vías a 0.25mm

### Detalle de Trazos

Grosor a 0.75 milésimas de pulgada

Separación entre trazos de 0.15 milésimas de pulgada

Terminación de cobre a 1 onza

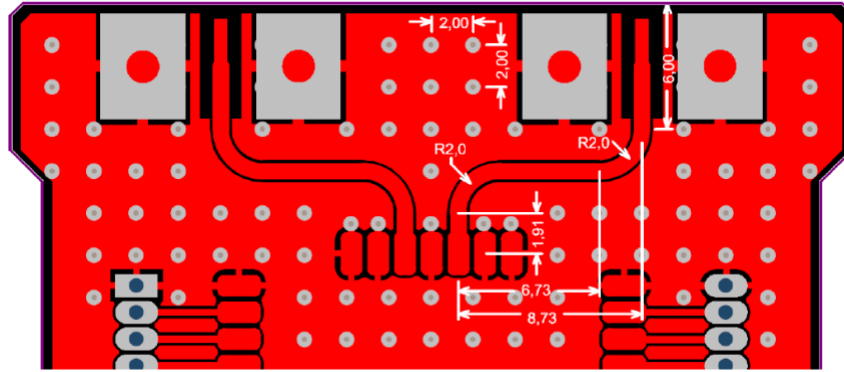


Figura 25. Layout de antenas LoRa.

Fuente: Microchip Technology Inc., 2017[58]

Las medidas están en milímetros.

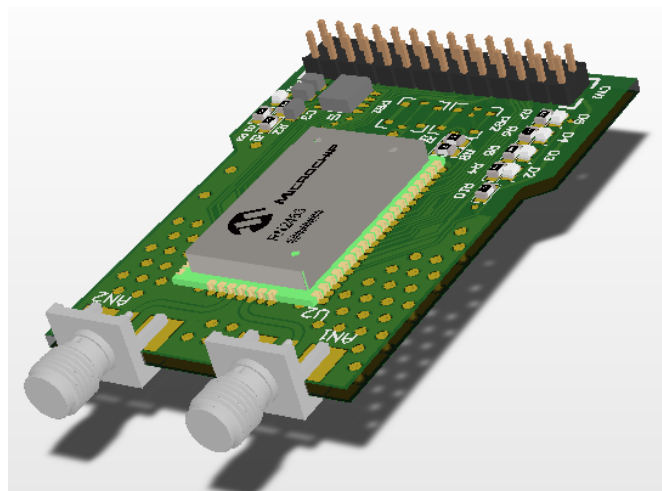


Figura 26. Imagen en 3D de la tarjeta LoRa.

Fuente: Elaboración propia.

#### 4.2.6 Diseño de Software

La tarjeta LoRa cuenta con una interfaz UART que es conectada al UART4 de la tarjeta BeagleBone Black, en la cual se diseñó un programa que valida la comunicación hacia la tarjeta, el funcionamiento de la tarjeta y la comunicación LoRa.

El módulo LoRa utiliza una sintaxis que divide sus comandos en tres tipos, que son de sistema, capa mac y capa de radio como se describe en el cuadro 26.

TIPO DE COMANDO	COMANDO	DESCRIPCIÓN
SISTEMA	SYS	SON COMANDOS QUE CONTROLAN EL COMPORTAMIENTO DEL SISTEMA, PUEDE ADQUIRIR INFORMACIÓN DEL FIRMWARE HARDWARE, O ACCEDER A LA MEMORIA EEPROM DEL MÓDULO.
LORA WAN CLASE A Y C	MAC	SON COMANDOS QUE PERMITEN MANIPULAR EL PROTOCOLO LORAWAN PARA COMPORTAMIENTO, ACCIONES Y CONFIGURACIONES.
TRANCEIVER	RADIO	CONTROLA LAS CONFIGURACIONES ESPECÍFICAS DEL RADIO.

*Cuadro 28. Tipos de comando de tranceiver RN2483A.*

*Fuente: Microchip, 2015[61].*

Todos los comandos utilizan parámetros en hexadecimal, además todos deben ser terminados por <CR><LF> y las réplicas serán terminadas de la misma manera.

Para la validación de la comunicación de UART, se utilizaron librerías proporcionadas por Adafruit que vienen precargadas en la plataforma BeagleBone Black y que ayudan a manipular diferentes periféricos de esta, en nuestro caso utilizamos las librerías del módulo UART y GPIO, que son las entradas y salidas de la tarjeta.

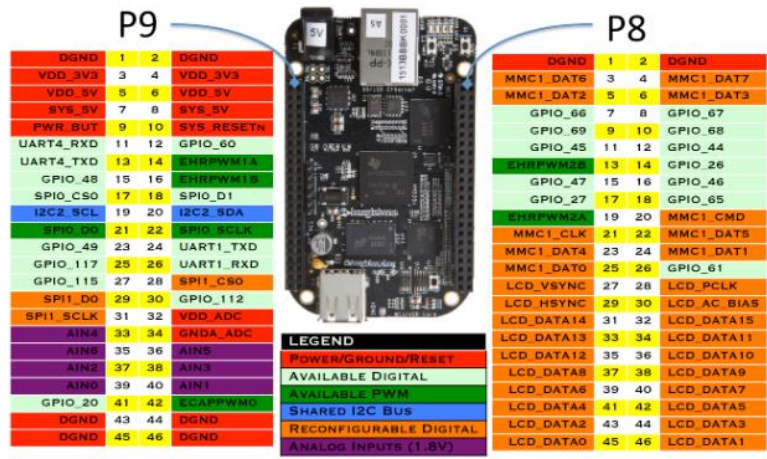


Figura 27. Pinout de tarjeta BeagleBone Black

Fuente: BeagleBoard.org, support, bone101<sup>33</sup>.

El programa configura el UART4 de la plataforma a una velocidad de 57600 baudios, 8 bits de transmisión, sin paridad y 1 bit de parada. Estos valores son utilizados de acuerdo con la configuración de fábrica del módulo.

El módulo cuenta con un pin de reset que nos permite reiniciar el módulo para evitar atascamiento en la comunicación, en nuestro caso lo aplicamos después de la configuración del UART para que el módulo reinicie antes de que se le envíe el primer comando. Enseguida se valida que el puerto serie “UART4” se encuentre disponible para ser utilizado.

Debido a que el trabajo presente se refiere a la validación del hardware del sistema la comunicación se establecerá por medio de la capa de radio del módulo para evitar crear un programa que utilice el protocolo LoRaWAN. LoRaWAN es un protocolo montado sobre la capa física LoRa, cuyo diseño de hardware permite la modulación de señal para alcanzar transmisiones de kilómetros con consumos de energía relativamente bajos. Dado esto se utilizó el comando “mac pause” para desactivar la capa mac y poder tomar el control de la capa física o de radio. Enseguida emitimos el comando “radio set pwr 14” que nos permite configurar la

<sup>33</sup> <https://beagleboard.org/Support/bone101>, BeagleBoard.org, open-hardware expandable computer, diciembre 15 de 2019.



potencia del radio de -3 a 15 dBm. Seguido de este comando enviamos la transmisión que como habíamos revisado se lanza con valores hexadecimales.

Como podemos observar el programa está leyendo todas las respuestas del módulo ya que cada vez que le enviamos un comando, este responde con un “ok” para indicar una recepción satisfactoria.

```
import serial
import Adafruit_BBIO.UART as UART
import Adafruit_BBIO.GPIO as GPIO
from time import sleep

loraReset = "P9_15"
GPIO.setup(loraReset,GPIO.OUT)

UART.setup("UART4")
ser=serial.Serial('/dev/ttyO4',57600,serial.EIGHTBITS,serial.PARITY_NONE,serial.STOPBITS_ONE,0)

def read():
    while ser.inWaiting() == 0:
        pass
    com = ser.readline()
    print com
def setPin(gpio,pinstate):
    command = 'sys set pindig'+ ' ' + gpio + ' ' + pinstate + '\r\n'
    ser.write(command)

ser.close()
ser.open()
GPIO.output(loraReset,GPIO.LOW)

sleep(1)
GPIO.output(loraReset,GPIO.HIGH)
read()

if ser.isOpen():
    print "Serial is Open"

ser.write('mac pause\r\n')
read()

ser.write('radio set pwr 14\r\n')
read()

ser.write('radio tx 686f6c61206d756e646f\r\n')
read()
read()
```

Figura 28. Programa de comunicación LoRa.

Fuente: Elaboración propia.

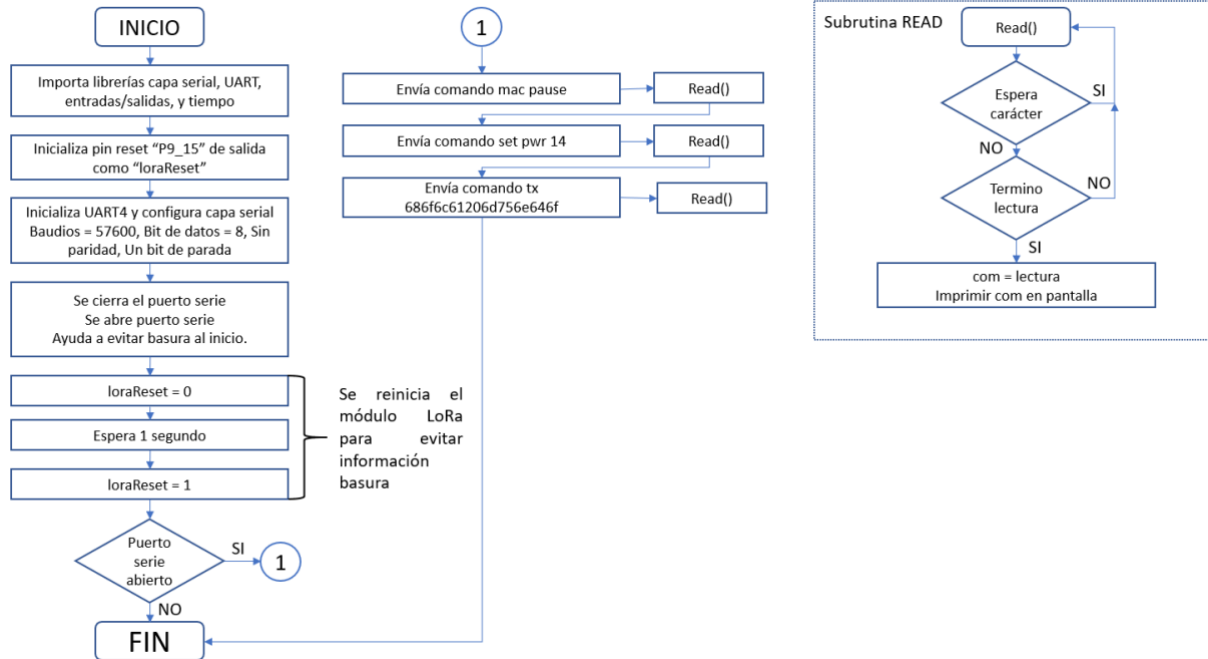


Figura 29. Diagrama de flujo de comunicación LoRa.

Fuente: Elaboración propia.

La Figura 28 muestra el código con el cual realizamos la comunicación LoRa, este está dividido en 3 partes, inicialización, configuración y ejecución. En inicialización se realiza toda la localización o llamadas a librerías que nos ayudan a ejecutar funciones útiles para el código, como las librerías de UART y entradas y salidas. En el caso de configuración, definimos todos los parámetros necesarios para que la capa física funcione, como es el caso del UART4, por otra parte, también se definen funciones de software que son usadas para facilitar la programación y reducir tamaño del código. Para finalizar en el código de ejecución se implementan los comandos necesarios para configurar el módulo LoRa para poder transmitir y recibir mensajes. En la Figura 29 podemos observar con mayor claridad el flujo del programa para su correcta ejecución.

```

root@beaglebone:/home/mobec/python# nano lora.py
root@beaglebone:/home/mobec/python# python lora.py
RN2483 1.0.4 Oct 12 2017 14:59:25

Serial is Open
4294967245

ok

ok

radio_tx_ok

```

Figura 30. Ejecución de programa de comunicación LoRa.  
Fuente: Elaboración propia.

La comunicación LoRa se realiza entre dos tarjetas con la misma capa física, es decir que se puedan entender entre ellas, en la Figura 30 se muestra la respuesta de la tarjeta transmisora en consola, el mensaje de “serial is open” está pensado para indicar al usuario que el puerto serial está libre para su uso y el programa puede continuar, el siguiente mensaje que se recibe como número, es la cantidad de segundos que la tarjeta LoRa mantendrá la capa MAC para poder utilizar la capa LoRa de manera directa, esto nos ayuda a evitar utilizar el protocolo LoRaWAN. Las siguientes respuestas de ok, las envía el módulo LoRa para indicar que el radio se configuró correctamente en mac pause y a una potencia de 14 dBm, El último mensaje de “radio\_tx\_ok” es para indicar que el mensaje fue transmitido de manera correcta.

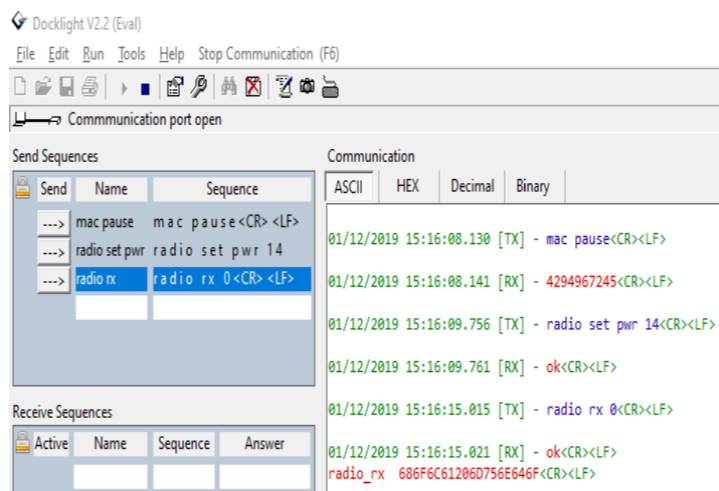


Figura 31. Comandos de recepción de comunicación LoRa.  
Fuente: Programa Docklight V2.2(Eval), elaboración propia.

Como lo mencionábamos anteriormente la comunicación LoRa se lleva a cabo entre dos dispositivos, el transmisor y receptor, en la Figura 29 y 30 explicamos el funcionamiento del transmisor. Para el caso del receptor utilizamos un programa llamado Docklight que nos permite emitir comandos de manera serial. Esto nos ayuda a simular un dispositivo de comunicación como lo es el Nodo Multiplataforma o la tarjeta BeagleBone Black. En la imagen 31 podemos observar que se utiliza la misma secuencia de comandos programados en la Figura 29 pero en vez de ejecutar el comando de transmisión, se aplica el comando “radio rx 0” el cual nos permite entrar en modo de recepción continua, es decir que todo el tiempo está listo para recibir mensajes del transmisor.

### 4.3 Resultados

Los resultados están basados en la correcta programación e integración de la tarjeta BeagleBone Black con la tarjeta LoRa A, esta integración se define como transmisor, asimismo, en su contraparte la tarjeta LoRa B con el programa Docklight, que nos ayuda a fungir como receptor. Las conexiones se muestran en la Figura 32.

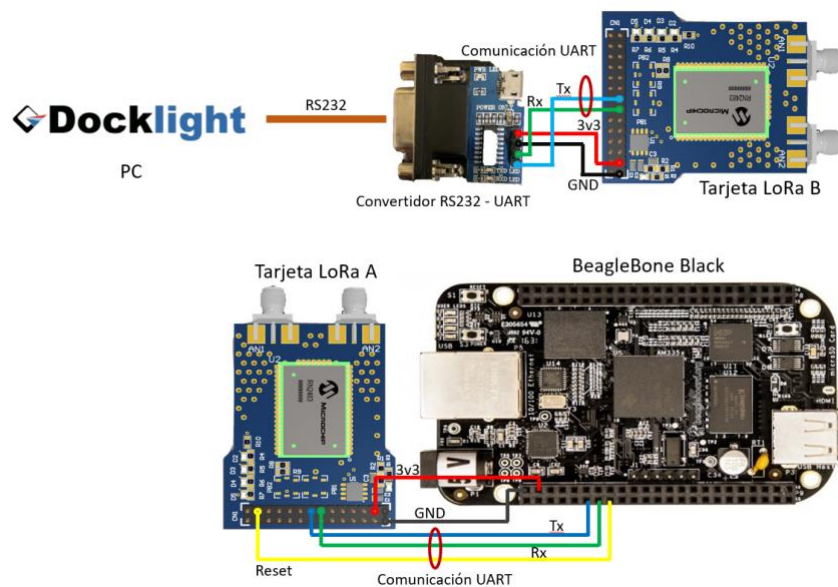


Figura 32 Acoplamiento de tecnologías.

Fuente: Elaboración propia.

Las pruebas de funcionalidad se rigen de acuerdo con el siguiente procedimiento.

**BeagleBone Black** – Tarjeta controladora del protocolo LoRa, envía comandos AT vía UART hacia la tarjeta LoRa. Estos comandos realizan configuraciones de la tarjeta y tranceiver, para que actúe como transmisor.

**Tarjeta Lora A** – Recibe comandos AT vía UART de la tarjeta BeagleBone Black, para determinar sus configuraciones como transmisor. Un comando de transmisión es enviado también para lanzar un paquete hacia la tarjeta LoRa B.

**Docklight** – Es un programa que ayuda a establecer comunicación entre una PC y un dispositivo externo por medio del protocolo de comunicación RS232. Aquí se configuran los comandos necesarios para enviarlos a la tarjeta LoRa B.

**Convertidor RS232/UART** – Este es un convertidor de señales serie RS232 a señales TTL. En este caso se utiliza para poder utilizar el protocolo UART.

**Tarjeta Lora B** – Recibe comandos AT vía UART de Docklight para determinar sus configuraciones como receptor continuo, para poder recibir lo que la tarjeta Lora A envíe por medio de protocolo LoRa.

Los resultados finales se pueden medir de acuerdo con la correcta comunicación entre estos dispositivos, que finalmente es lo que se busca para validar la integración de tecnologías propuesta en este trabajo. Cabe destacar que esta prueba solo fue la validación de comunicación, lo cual nos permite garantizar el funcionamiento de hardware, sin embargo, con un desarrollo de software mas elaborado se puede llegar a la implementación de prototipos funcionales.



## Capítulo 5

# Impacto Social y Económico

## Capítulo 5. Impacto Social y Económico

### 5.1 Socios claves

El nodo multiplataforma de conectividad para IoT tiene la ventaja de ser un Gateway dedicado a cualquier tipo de red del internet de las cosas que cubra los protocolos WiFi, USB, LoRa y Bluetooth, estos protocolos en su mayoría son de uso común en la vida cotidiana de las personas, por lo que existe un amplio mercado para el mismo.

PROTOCOLO	MERCADO	DESCRIPCIÓN	OBSERVACIÓN
WIFI	TECNOLOGÍAS CON NECESIDAD DE CONEXIÓN INALÁMBRICA DE REDES DE ÁREA LOCAL. LA TECNOLOGÍA LA PODREMOS UBICAR EN CUALQUIER SISTEMA DEL INTERNET DE LAS COSAS.	LA TECNOLOGÍA ESTÁ LIMITADA A SER APLICADA EN ENTORNOS INSTITUCIONALES Y DOMÉSTICOS, CON NECESIDADES DE ALCANCE NO MAYORES A LOS 200 METROS	ES POR ESO POR LO QUE EN LA ARQUITECTURA DEL NODO MULTIPLATAFORMA SE PROPONE COMO ENLACE ENTRE EL SISTEMA Y EL SERVIDOR DE INTERNET, ADEMÁS DE QUE EL PROPÓSITO DE LA TECNOLOGÍA VA ENFOCADO A ESTE USO.
BLUETOOTH	TECNOLOGÍAS CON LA NECESIDAD DE CONEXIÓN DE DISPOSITIVOS MÓVILES CON ENFOQUE A LA TRANSMISIÓN DE DATOS Y VOZ A DISTANCIAS CORTAS. MERCADOS DE USO PERSONAL, LA MAYORÍA DE LOS DISPOSITIVOS PERSONALES CUENTAN CON ESTA TECNOLOGÍA.	ESTA TECNOLOGÍA ESTÁ ORIENTADA AL USO PERSONAL.	EN SU MAYORÍA SE USA PARA LA TRANSFERENCIA DE DATOS Y VOZ. CON ESTA TECNOLOGÍA LOGRAMOS ABARCAR EL MERCADO PERSONAL Y DE USO DIARIO.
USB	ES UNA TECNOLOGÍA ALÁMBRICA APLICADA PARA LA TRANSFERENCIA DE DATOS ENTRE DOS DISPOSITIVOS EN SU MAYORÍA PERIFÉRICOS.	ESTA TECNOLOGÍA PUEDE SER UTILIZADA PARA CONECTAR VARIOS PERIFÉRICOS EN UNA SOLA TERMINAR, PARA TRANSFERENCIA DE DATOS Y ALMACENAMIENTO.	SE AGREGA AL NODO MULTIPLATAFORMA PARA ADICIONAR PUNTOS DE ALMACENAMIENTO Y ACCESO AL SISTEMA. ES DE GRAN UTILIDAD PARA EL USUARIO.
LORA	ES DE LAS TECNOLOGÍAS MÁS RECIENTES EN APLICACIONES AL MERCADO ABIERTO Y SUS CARACTERÍSTICAS DE BAJO CONSUMO DE ENERGÍA CON LARGO ALCANCE, LA HACE LA HERRAMIENTA PERFECTA PARA IMPLEMENTACIÓN DE REDES DE IOT QUE NECESITEN RANGOS DE COMUNICACIÓN DE KILÓMETROS.	ES UNA TECNOLOGÍA INALÁMBRICA DE BAJO CONSUMO DE ENERGÍA QUE PROPORCIONA RANGOS DE COMUNICACIÓN MAYORES A LOS 4 KILÓMETROS DE DISTANCIA.	NOS AYUDA A ABARCAR MERCADOS DE CIUDADES INTELIGENTES, ENERGÍA, AGRICULTURA Y MÁS.

Cuadro 29. Análisis de mercados por tecnología de comunicación del Nodo Multiplataforma.

Fuente: Elaboración propia.

Como podemos observar el Nodo multiplataforma es una herramienta con bastante potencial de desarrollo sobre cualquier red del IoT, el desarrollo de este Gateway potencializa la mayoría de los mercados del internet de las cosas como, ciudades inteligentes, energía, agricultura, agroindustria, industria 4.0, y muchos más, y dado a que este es una solución aplicada a la mayoría de estos mercados , por lo que lo socios potenciales para el desarrollo, implementación y comercialización son los proveedores de los componentes más importantes de la plataforma y las empresas que encabezan el IoT en cada uno de los rubros.

## **5.2 Actividades clave**

Aunque el nodo multiplataforma sea una tarjeta con bastantes prestaciones que permiten ser ingresado a cualquier mercado del IoT, es necesario realizar ciertos pasos que nos lleven al éxito comercial del mismo.

1. La definición de una marca que permita transmitir que el producto es de alto valor para la implementación de cualquier red del internet de las cosas, sin importar el rubro o área de esta.
2. Formar una relación de proveedores valiosos, que en nuestro caso es la de los fabricantes de los componentes principales de la tarjeta, ya que estos pueden proveer de información e ingeniería valiosa que permitan lanzar al mercado el producto más rápido y con mayor visibilidad, aprovechando el prestigio de estos.
3. La calidad del producto aplicando las mejores prácticas de diseño y desarrollo de hardware, lo cual nos lleva a una reducción de costos y tiempo de manufactura.
4. La fuerza de venta de este tipo de productos se da implementando sistemas exitosos que demuestren la versatilidad y eficiencia del producto.
5. En el caso de la comercialización se opta por la venta en línea para la creación de distribuidores autorizados del producto.



6. El servicio al cliente se dará a través de los distribuidores ofreciéndoles capacitación para el desarrollo de sistemas basados en el Nodo Multiplataforma de Conectividad para IoT.

### 5.3 Recursos claves

El nodo multiplataforma de conectividad para IoT entra en el mercado de la fabricación y comercialización de productos. Actualmente se encuentra en la “fase III Desarrollo de Hardware”, del ciclo de desarrollo de un sistema embebido [62].

Tomando este como punto de partida los recursos clave se pueden apreciar de la siguiente manera.

ETAPA	RECURSO
DESARROLLO DE HARDWARE	COMPUTADORA DE ALTO RENDIMIENTO.
	LICENCIA DE PROGRAMA DE DISEÑO DE HARDWARE.
	ESPECIALISTA EN DISEÑO DE HARDWARE Y TARJETAS DE CIRCUITO IMPRESO.
	FABRICACIÓN DE PROTOTIPO.
DESARROLLO DE SOFTWARE	ESPECIALISTA EN DISEÑO DE SOFTWARE EMBEBIDO
INTEGRACIÓN Y PRUEBAS	LABORATORIO DE DISEÑO CON MATERIALES DE ANÁLISIS DE FUNCIONAMIENTO DE TARJETAS DE CIRCUITO IMPRESO.
ADMINISTRACIÓN DEL PRODUCTO	ESPECIALISTA EN ADMINISTRACIÓN
MANUFACTURA EN SERIE	PROVEEDOR DE FABRICACIÓN

*Cuadro 30. Recursos Clave para la fabricación del Nodo Multiplataforma.*

*Fuente: Elaboración propia.*

### 5.4 Propuesta de valor

El nodo multiplataforma de conectividad para IoT es un Gateway que integra protocolos de comunicación de alto impacto sobre el IoT, lo que lo convierte en un dispositivo capaz de ser colocado en la arquitectura de la mayoría de los mercados de este, esta versatilidad lo hace bastante atractivo para ser integrado en cualquier red que necesite acceso a internet, periféricos, almacenamiento, transferencia de voz y datos, y transmisiones de largo alcance.

El integrar estos cuatro protocolos en una tarjeta, con un procesador capaz de manejarlos todos al mismo tiempo, que procesa la información para ser enviada al servidor, es una aportación de alto valor hacia el cliente, usuario o sistema ya que facilita su uso e integración.

El problema de integrar diferentes dispositivos a una red, debido a que cada uno cuenta con un protocolo distinto y único, se resuelve embebiendo los mismos en una sola tarjeta o plataforma, reduciendo costos, espacio y tiempos de instalación.

## **5.5 Clientes**

Como lo hemos revisado, actualmente existe una enorme variedad de clientes para el Nodo Multiplataforma de Conectividad para IoT, hemos mencionado la mayoría de los mercados del internet de las cosas y además puede ser utilizado como plataforma de investigación, desarrollo, educación y muchos más. El microprocesador que lo integra cuenta con un microcontrolador interno que facilita la implementación de sistemas embebidos de control y manipulación de actuadores.

## **5.6 Costos**

Los costos obtenidos en el presente trabajo están en dólares americanos y con cotización de compra a una pieza por componente, además solo se consideraron componentes de alto nivel, omitiendo algunos conectores y componentes pasivos.

COMPONENTE	DESCRIPCIÓN	COS TO EN USD	CANTIDAD	SUBTOTAL	PROVEEDOR
WM6698CT-ND	MICROSD CARD CONNECTOR	\$1.16	1.00	\$1.16	DIGIKEY
757-THGAMRG7T13BAIL	EMMC 16GB EMMC 5.1	\$14.91	1.00	\$14.91	MOUSER
AS4C256M16D3LB-12BCN	DDR3L	\$9.79	1.00	\$9.79	DIGIKEY
WL1835MODGBMOCT	BLUETOOTH, WIFI 802.11B/G/N, BLUETOOTH 4.2 TRANSCEIVER MODULE 2.4GHZ MONTAGE EN SUPERFICIE	\$17.16	1.00	\$17.16	DIGIKEY
RN2483A-I/RM104-ND	MODULO LORA	\$12.80	1.00	\$12.80	DIGIKEY
87520-0010BLF	USB HOST	\$0.73	1.00	\$0.73	DIGIKEY
UX60SC-MB-5ST (80)	USB MINI B	\$0.80	1.00	\$0.80	DIGIKEY
AM3358BZCZ100	PROCESADOR	\$17.21	1.00	\$17.21	DIGIKEY
296-21569-1-ND	REGULADOR LOW DROP OUT	\$1.13	1.00	\$1.13	DIGIKEY
CC0805KX7R5BB105	CAPACITOR DE 1UF	\$0.26	1.00	\$0.26	DIGIKEY
732-7646-1-ND	CAPACITOR DE 2.2UF	\$0.19	1.00	\$0.19	DIGIKEY
GRM216R71H471KA01J	CAPACITOR DE 470 PF	\$0.12	1.00	\$0.12	DIGIKEY
A126370CT-ND	RESISTENCIA DE 470K	\$0.10	1.00	\$0.10	DIGIKEY
311-280KCRCT-ND	RESISTENCIA DE 280K	\$0.10	1.00	\$0.10	DIGIKEY
475-1410-1-ND	LED VERDE	\$0.27	5.00	\$1.35	DIGIKEY
A129741CT-ND	RESISTENCIAS 220 OHM	\$0.10	5.00	\$0.50	DIGIKEY
RR12P10.0KDCT-ND	RESISTENCIAS DE 10K	\$0.11	1.00	\$0.11	DIGIKEY
P8047SCT-ND	BOTONES	\$0.38	2.00	\$0.76	DIGIKEY
FABRICACIÓN Y ENSAMBLE	FABRICACIÓN Y ENSAMBLE DE LA TARJETA	\$30.00	5.00	\$150.00	N/A
<b>TOTAL</b>				<b>\$229.18</b>	

Cuadro 31. Costos de materiales del Nodo Multiplataforma.

Fuente: Elaboración propia.

La estimación del costo de la fabricación y ensamble de la tarjeta se estimará con base a los costos obtenidos en el desarrollo de la tarjeta LoRa, el proveedor fábrica como mínimo cinco tarjetas, pero solo se está considerando ensamblar una.



## Conclusiones

## Conclusiones

El desarrollo del presente trabajo se piensa a raíz del incremento constante de dispositivos conectados a redes del IoT. Es importante que se comprenda que el hablar de internet no quiere decir que debe de ser público o de acceso libre, si no el de tener un servidor que nos permita manipular el sistema de manera remota. Estos dispositivos conectados a internet o servidor incrementan año con año de manera exponencial, y sus redes tienen la necesidad de ser controladas de manera eficiente para ser direccionadas por una puerta de enlace o Gateway, la cual va de la capa de red de dispositivos hasta la de aplicación de usuario.

El “Nodo Multiplataforma de Conectividad para IoT” brinda una solución adecuada a este problema ya que emplea tecnologías de comunicación de alto impacto sobre el uso cotidiano del internet. Existen varias soluciones en el mercado que cuentan con este enfoque, sin embargo, la diferenciación sobre estas es la integración del protocolo LoRa, ya que este nos permite abarcar un mayor número de mercados o áreas del internet de las cosas, sobre todo en los que se necesite una comunicación inalámbrica a distancias grandes, es decir, que requieran un rango de alcance en kilómetros. Siendo LoRa un dispositivo que puede abarcar hasta 10 kilómetros a la redonda, algunas áreas o mercados de aplicación son, las ciudades inteligentes, en aplicaciones como la sincronización de semáforos de cruce a cruce, control de lámparas de manera remota, localización vehicular, y todas estas que requieren no tener cableado. LoRa es un protocolo versátil de bajo consumo energético y de bajo costo, que además de abarcar grandes distancias, nos permite ser implementado también en aplicaciones que no las necesiten o en rangos menores, como pueden ser edificios y casas inteligentes, tiendas departamentales, etc.

El objetivo principal del trabajo es la integración de hardware de alto nivel que está basado en tecnologías previamente probadas y analizadas para la correcta implementación de la plataforma, además se realiza el diseño de un módulo LoRa de fácil acoplamiento para un rápido desarrollo de prototipo y pruebas, todo esto

basado en las mejores prácticas de integración de hardware y diseño de tarjetas de circuito impreso.

Los protocolos de comunicación fueron especialmente seleccionados para que la plataforma sea de fácil integración en las redes del internet de las cosas sin importar la arquitectura del sistema y su aplicación, ya que este es un Gateway o puerta de enlace que su función principal es la de recibir y transmitir información de sus protocolos, para que la aplicación de usuario pueda procesarlos.

El impacto de la plataforma es el de ser un recurso de alto valor para el empleo del internet de las cosas que le ayude a soportar en sus redes el despliegue de cientos de dispositivos conectados al sistema a costos bajos.

Como implementaciones futuras, en este trabajo se desarrolló la integración de hardware a un nivel de componentes de mayor impacto en la plataforma, si continuamos con el esquema de procesos de desarrollo de sistemas embebidos estamos situados en la fase de desarrollo de hardware, a un 50% de avance. Los trabajos futuros los guiaremos con base a este esquema y los dividiremos prototipo funcional y producto.

- a) Para llegar a un prototipo funcional es necesario realizar varias iteraciones en el diseño. Que son desarrollo de hardware, software y pruebas.

Para el desarrollo de hardware, se deben definir los componentes pasivos de cada una de las tecnologías seleccionadas en el trabajo. Se procede al desarrollo de esquemáticos y tarjeta de circuito impreso, para este es necesario seguir con las mejores prácticas propuestas por IPC-2222.

En el desarrollo de software es necesario continuar con la distribución del sistema operativo junto con la instalación de controladores y desarrollo de aplicaciones.

Para las pruebas es necesario probar el hardware a niveles eléctricos y de temperatura, en software es la manipulación del hardware a beneficio del usuario o del sistema.

- b) Para la obtención del producto es necesario contar con un prototipo funcional que nos permita continuar a la validación de la tecnología por medio de patentes, certificaciones, y comercialización de esta.

Para mitigar el retraso de lanzamiento es recomendable contar con un equipo de especialistas en el área de hardware y software en la fase de prototipo, sin embargo, este desarrollo prueba el correcto funcionamiento de la plataforma por lo que el prototipo y fase de producto se pueden trabajar en paralelo.

## Bibliografía

- [1] V. Alcázar, “Estudio de las prestaciones de IEEE 802.15.4e (Tesis de pregrado).”, p. 66, 2013.
- [2] L. Fedele and L. Fedele, “From Basic Maintenance to Advanced Maintenance,” *Methodol. Tech. Adv. Maint.*, pp. 63–112, 2011.
- [3] S. Li, L. Da Xu, and S. Zhao, “The internet of things: a survey,” *Inf. Syst. Front.*, vol. 17, no. 2, pp. 243–259, 2015.
- [4] D. Evans, “Internet de las cosas Internet de las cosas Cómo la próxima evolución de Internet lo cambia todo,” p. 11, 2011.
- [5] T. Zachariah, N. Klugman, B. Campbell, J. Adkins, N. Jackson, and P. Dutta, “The internet of things has a gateway problem,” *HotMobile 2015 - 16th Int. Work. Mob. Comput. Syst. Appl.*, pp. 27–32, 2015.
- [6] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,” *IEEE Commun. Surv. Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.
- [7] “IoT e Industria 4.0: ¿cuál es la diferencia? | noticias.ltda.” [Online]. Available: <https://www.noticias.ltda/sociedad-digital/iot-e-industria-40-diferencia/>. [Accessed: 22-Jun-2020].
- [8] S. D. E. Máquinas and Y. G. D. E. Datos, “Escuela politécnica de ingeniería de gijón.”, 2019.
- [9] A. Jayaram, “Smart Retail 4 . 0 IoT Consumer Retailer Model for Retail Intelligence and Strategic Marketing of In-store Products Athul Jayaram Amity University Uttar Pradesh Keywords,” *17th Int. Bus. Horizon-INBUSH ERA-2017*, no. February, 2017.
- [10] J. C. Vela-Medina, A. E. Guerrero-Sanchez, J. E. Rivas-Araiza, and E. A. Rivas-Araiza, “Face detection for efficient video-surveillance IoT based embedded system,” *IEEE ICA-ACCA 2018 - IEEE Int. Conf. Autom. Congr. Chil. Assoc. Autom. Control Towar. an Ind. 4.0 - Proc.*, pp. 1–6, 2019.
- [11] S. Barillaro *et al.*, “Diseño de sistema IoT de monitoreo y alarma para personas mayores,” *XVIII Work. Investig. en Ciencias la Comput.*, vol. 1, pp. 712–716, 1980.
- [12] J. A. Guerrero-Ibañez *et al.*, “SGreenH-IoT: Plataforma IoT para Agricultura de Precisión.”, *CISCI 2017 - Decima Sexta Conf. Iberoam. en Sist. Cibern. e Informatica, Decimo Cuarto Simp. Iberoam. en Educ. Cibern. e Informatica, SIECI 2017 -*



*Memorias*, pp. 315–320, 2017.

- [13] DPA and M. Martínez Anarozco, “DPA (2015) Los semáforos inteligentes. National Geographic , 1 , 2.,” *Natl. Geogr. Mag.*, vol. 1, p. 15, 2015.
- [14] R. Alejandro, “Universidad Politécnica Salesiana Sede Quito,” *Tesis*, vol. 04, pp. 1–100, 2019.
- [15] M. T. De Grado, “DISEÑO E IMPLEMENTACIÓN DE UN DEL AIRE Y VOLÚMENES DE EXPLOTACIÓN USANDO IOT Y VISIÓN POR COMPUTADOR , PARA APOYAR LABORES DE FISCALIZACIÓN,” 2019.
- [16] X. Chen, J. Liu, X. Li, L. Sun, and Y. Zhen, “Integration of IOT with smart grid,” *IET Conf. Publ.*, vol. 2011, no. 586 CP, pp. 723–726, 2012.
- [17] M. U.Farooq, M. Waseem, S. Mazhar, A. Khairi, and T. Kamal, “A Review on Internet of Things (IoT),” *Int. J. Comput. Appl.*, vol. 113, no. 1, pp. 1–7, 2015.
- [18] “Pros y contras de la metodología en cascada | OBS Business School.” [Online]. Available: <https://obsbusiness.school/es/blog-project-management/metodologia-agile/pros-y-contras-de-la-metodologia-en-cascada>. [Accessed: 22-Jun-2020].
- [19] I. Publishing, “S e n s o r s & T r a n s d u c e r s A Design of the IOT Gateway for Agricultural,” *Sensors and Transducers*, vol. 172, no. 6, pp. 75–80, 2014.
- [20] G. Aloï *et al.*, “A mobile multi-technology gateway to enable IoT interoperability,” *Proc. - 2016 IEEE 1st Int. Conf. Internet-of-Things Des. Implementation, IoTDI 2016*, pp. 259–264, 2016.
- [21] E. Gioia, P. Passaro, and M. Petracca, “AMBER: An advanced gateway solution to support heterogeneous IoT technologies,” *2016 24th Int. Conf. Software, Telecommun. Comput. Networks, SoftCOM 2016*, 2016.
- [22] D. Min, Z. Xiao, B. Sheng, H. Quanyong, and P. Xuwei, “Design and implementation of heterogeneous IOT gateway based on dynamic priority scheduling algorithm,” *Trans. Inst. Meas. Control*, vol. 36, no. 7, pp. 924–931, 2014.
- [23] “Ejemplos de Computadoras Embebidas | Lanner.” [Online]. Available: <https://www.lanner-america.com/es/blog-es/ejemplos-de-computadoras-embebidas/>. [Accessed: 11-Nov-2019].
- [24] A. Científico and I. N. G. O. Oña, “Diseño de un sistema de video vigilancia IP y alarma basada en movimiento , utilizando software libre sobre un computador de placa reducida , para la empresa color 2000 de la ciudad de Ibarra,” pp. 1–15, 2018.
- [25] H. Uriel and V. Gonzalez, “Microprocesadores Y Microcontroladores,” p. 295, 2009.
- [26] R. Pedruelo, “El Procesador de Señales.”

- [27] M. U. Martinez, *Arquitectura del PC. Volumen I: Microprocesadores*. 2003.
- [28] C. M. F. Marcos Serrano Carrillo, *Mantenimiento de equipos informáticos*. 2019.
- [29] E. Signal, I. Ethernet, D. Modes, D. S. Sequencing, and P. D. Switch-on, “AM335x Sitara™ Processors 1,” 2014.
- [30] F. Semiconductor, “Data Sheet : Technical Data Applications Processors,” pp. 1–167, 2012.
- [31] M. A. Processor, “Allwinner A64 Datasheet,” pp. 1–3, 2015.
- [32] Andrew s. Tanenbaum, *Sistemas Operativos Modernos*. 2003.
- [33] W. Stallings, *Multiprocessor and Real-Time Scheduling*. 2012.
- [34] D. International, “U-Boot Reference Manual.”
- [35] “Todo-Redes.” [Online]. Available: <https://todo-redes.com/equipos-de-redes/gateway-puerta-de-enlace>. [Accessed: 03-Dec-2019].
- [36] I. Yaqoob, I. A. T. Hashem, A. Ahmed, S. M. A. Kazmi, and C. S. Hong, “Internet of things forensics: Recent advances, taxonomy, requirements, and open challenges,” *Futur. Gener. Comput. Syst.*, vol. 92, pp. 265–275, 2019.
- [37] E. L. E. leee, “3.1 introducción,” pp. 36–51.
- [38] “Our Brands | Wi-Fi Alliance.” [Online]. Available: <https://www.wi-fi.org/who-we-are/our-brands>. [Accessed: 22-Jun-2020].
- [39] R. Castro, “Avanzando en la seguridad de las redes WIFI Going forward more Secure WIFI Networks,” *boletín la Red Nac. I+D RedIRIS*, no. N°. 73, pp. 23–32, 2005.
- [40] G. Lorefice Sparacino, “Disponible en: <http://www.redalyc.org/articulo.oa?id=78420205>,” *Rev. Electrónica Estud. Telemáticos - Tecnol. Ina. Bluetooth sobre los Serv. Comun. en los ámbitos Soc. y Empres.*, vol. 2, no. 2, pp. 36–49, 2003.
- [41] A. Rodríguez Penin, “Comunicaciones industriales. Guía práctica: Sistemas de regulación y control automáticos,” vol. 03, p. 290, 2009.
- [42] B. Reynders and S. Pollin, “Chirp spread spectrum as a modulation technique for long range communication,” *2016 IEEE Symp. Commun. Veh. Technol. Benelux, SCVT 2016*, no. 2, pp. 0–4, 2016.
- [43] leee, *IEEE Std 802.15.4a-2007, IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-LANs and MANs-Specific Requirements-Part 15.4: Wireless MAC and PHY Specifications for LR-WPANs-Amendment 1: Add Alternate PHYs*, vol. 2007, no. August. 2007.
- [44] T. M. Workgroup, “A technical overview of LoRa ® and LoRaWAN™ What is it?,” no.

November, 2015.

- [45] A. Patrizi, P. L. Mattioli, L. Lugatti, and P. Patrone, "Malattie Bollose: Studio Epidemiologico Su 164 Pazienti in Eta Adulta," *G. Ital. di Dermatologia e Venereol.*, vol. 127, no. 10, pp. 479–483, 1992.
- [46] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, "Future internet: The internet of things architecture, possible applications and key challenges," *Proc. - 10th Int. Conf. Front. Inf. Technol. FIT 2012*, pp. 257–260, 2012.
- [47] Allwinner, "Allwinner H3 Datasheet v1.2," no. February, 2007.
- [48] ARM, "ARM Architecture Reference Manual," pp. 1–1138, 2007.
- [49] "Microprocessor Cores and Technology – Arm." [Online]. Available: <https://www.arm.com/products/silicon-ip-cpu>. [Accessed: 10-Jun-2020].
- [50] Silicon labs, "Which ARM Cortex Core Is Right for Your Application : A , R or M ?," *Silicon Labs*, pp. 1–5, 2002.
- [51] H. D. Timeline, "AM335x Hardware Design Guide Selecting the Boot Mode."
- [52] Texas Instruments, "Technical Reference Manual AM335x and AMIC110 Sitara™ Processors," vol. 49, no. 0, pp. 1–148, 2009.
- [53] P. No, B. Black, and C. Page, "NOTE : PCB Revision for this board is Rev B6 IN WRITING THE COPYRIGHT HOLDERS AND / OR OTHER PARTIES," pp. 1–11, 2014.
- [54] "SD Standard Overview - SD Association." [Online]. Available: <https://www.sdcard.org/developers/overview/>. [Accessed: 08-Dec-2019].
- [55] D. D. Rate and S. Jesd, "Double Data Rate ( DDR ) SDRAM lu ci on es In d us tr ia le s lu on es In d tr ia le s," no. May 2005, 2008.
- [56] B. Mpn, "BeagleBone Black Wireless with WiFi and Bluetooth."
- [57] "Giving Linux the Boot," no. July, 2012.
- [58] Microchip Technology Inc., "Low-Power Long Range LoRa® Technology Transceiver Module," pp. 1–22, 2017.
- [59] M. Day and T. Instruments, "Understanding Low Drop Out ( LDO ) Regulators," pp. 1–7, 2019.
- [60] A. O. Voltage, E. Tight, R. Over, L. T. Coefficient, C. Limiting, and T. Protection, "500-mA LOW-NOISE LOW-DROPOUT VOLTAGE REGULATOR TL5209 500-mA LOW-NOISE LOW-DROPOUT VOLTAGE REGULATOR," no. May, 2007.
- [61] Microchip, *RN2483 LoRa™ Technology Module Command Reference User's Guide*. 2015.

[62] J. Carlos and T. Mosqueda, *Diagnóstico para la fundamentación de la Maestría en Sistemas Embebidos*. .