

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería Electrónica



**Optimización del sistema de caracterización de parámetros climáticos en mini
jardines clonales forestales bajo ambiente protegido**

**Informe de Proyecto de Graduación para optar por el título de Ingeniero en
Electrónica con el grado académico de Licenciatura**

Frank Román Muñoz

Cartago, Junio de 2012

INSTITUTO TECNOLÓGICO DE COSTA RICA

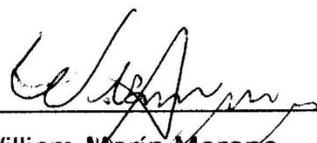
ESCUELA DE INGENIERÍA ELECTRÓNICA

PROYECTO DE GRADUACIÓN

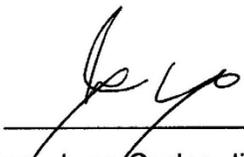
TRIBUNAL EVALUADOR

Proyecto de Graduación defendido ante el presente Tribunal Evaluador como requisito para optar por el título de Ingeniero en Electrónica con el grado académico de Licenciatura, del Instituto Tecnológico de Costa Rica.

Miembros del Tribunal


Ing. William Marín Moreno

Profesor lector


Ing. Juan Carlos Jiménez Robles

Profesor lector


Ing. Johan Carvajal Godínez

Profesor asesor

Los miembros de este Tribunal dan fe de que el presente trabajo de graduación ha sido aprobado y cumple con las normas establecidas por la Escuela de Ingeniería Electrónica

Cartago, junio 2012

Declaratoria de autenticidad

Declaro que el presente Proyecto de Graduación ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos propios.

En los casos en que he utilizado bibliografía, he procedido a indicar las fuentes mediante las respectivas citas bibliográficas.

En consecuencia, asumo la responsabilidad total por el trabajo de graduación realizado y por el contenido del correspondiente informe final.

Cartago, Junio 2012



Frank Román Muñoz

Céd: 3-0441-0533

Resumen

Este documento describe el proceso de investigación y optimización del sistema de caracterización de parámetros climáticos en mini jardines clonales implementado en un invernadero de GENFORES¹ ubicado en el Instituto Tecnológico de Costa Rica con sede en Santa Clara, San Carlos. Con la puesta en marcha de este proyecto se pondrá contar con un sistema que permite comunicar la información obtenida desde el lugar donde se encuentran los nodos con sus sensores, hasta cualquier punto donde se tenga acceso a internet.

Los procedimientos seguidos para la ejecución del proyecto se basaron en un primer diagnóstico realizado en el sitio evaluando las principales deficiencias del sistema implementado en primer lugar. Dicho diagnóstico mostró la necesidad de implementar cambios a la hora de transmitir la información desde el nodo base de la red hasta el punto de acceso para el usuario. Para ello se rediseñó la arquitectura de la red implementando una tarjeta Beaglebone encargada de controlar un módem USB, el cual mediante la tecnología de tercera generación permite el acceso a internet. Para transmitir la información fue necesario emplear el protocolo de comunicación I2C entre el nodo base y la tarjeta Beaglebone. También se requirió añadir un módulo adicional de iluminación el cual permite extender el fotoperiodo de las plantas como parte de los experimentos de GENFORES.

La optimización implementada en las rutinas de software permitió cumplir con los requerimientos de disminuir el consumo de corriente para evitar el sobrecalentamiento en el nodo base. Todos los objetivos alcanzados son importantes por su contribución con la labor del equipo de GENFORES en la búsqueda del mejoramiento de los recursos genéticos forestales.

Palabras claves: Red de sensores, Beaglebone, Módem USB, Protocolo I2C, Multimedia Board, Encapsulado de sensores.

¹ GEN=genética, FORES=bosque. Cooperativa especialista en mejoramiento Genético Forestal.

Abstract

This document describes the process of research and optimization of the system for characterization of weather parameters in mini clonal forest gardens under controlled environment conditions implemented at a GENFORES² green house located in Santa Clara, in the Costa Rica Institute of Technology. With the implementation of the project, it will be able to send the recollected information from the sensors through the nodes and upload it to internet.

The procedures for the implementation of the project were based on a first assessment of the case done in the location to diagnose the main deficiencies of the employed system one year ago. The results identified the need to implement changes to find a different way to transport the information from the base node to the user access point. For this reason, the architecture of the network was redesigned using a Beaglebone to control a USB modem, which allows the internet access through third generation technology. The I2C communications protocol was used to transmit information between the base node and the Beaglebone. Also, it was necessary to add an additional module of lighting, which allows the extending of the plants' photoperiod as part of the GENFORES experiments.

The optimizations implemented in the software routines allowed to reach the requirements in the decrease of current consume to avoid overheating on the base node. All the reached objectives are important because their contribution with the GENFORES team labor to ensure the improvement and development of forest genetic resources.

Keywords: Sensor Networks, Beaglebone, USB Modem, I2C Protocol, Multimedia Board, Sensor package.

² GEN=genetic, FORES=forest. Cooperative specialist in forest genetic improvement.

Índice general

ÍNDICE DE FIGURAS	9
ÍNDICE DE TABLAS	10
CAPÍTULO 1: INTRODUCCIÓN	11
1.1 PROBLEMA EXISTENTE E IMPORTANCIA DE SU SOLUCIÓN	12
1.1.1 <i>Definición del problema</i>	12
1.1.2 <i>Síntesis del problema</i>	14
1.1.3 <i>Importancia de la solución</i>	14
1.2 SOLUCIÓN SELECCIONADA.....	14
CAPÍTULO 2: META Y OBJETIVOS.....	16
2.1 META.....	16
2.2 OBJETIVO GENERAL	16
2.3 OBJETIVOS ESPECÍFICOS	16
CAPÍTULO 3: MARCO TEÓRICO	17
3.1 DESCRIPCIÓN GENERAL DEL SISTEMA A MEJORAR	17
3.2 ANTECEDENTES BIBLIOGRÁFICOS	18
3.3 DESCRIPCIÓN DE LOS PRINCIPALES PRINCIPIOS FÍSICOS Y/O ELECTRÓNICOS RELACIONADOS CON LA SOLUCIÓN DEL PROBLEMA	18
3.3.1 <i>Redes de sensores</i>	18
3.3.2 <i>Sensores</i>	20
3.3.3 <i>Sensores de temperatura</i>	20
3.3.4 <i>Sensores de Humedad Relativa</i>	20
3.3.5 <i>Sensores de luz</i>	21
3.3.6 <i>PIC 32 Starter kit</i>	21
3.3.7 <i>Miwi</i>	23
3.3.8 <i>Modem USB</i>	24
3.3.9 <i>PPP (Protocolo punto a punto)</i>	25
3.3.10 <i>Beaglebone</i>	27
3.3.11 <i>Protocolo de comunicación I²C</i>	28
3.3.12 <i>Multimedia Board PIC32MX</i>	29
CAPÍTULO 4: PROCEDIMIENTO METODOLÓGICO	31
4.1 INVESTIGACIÓN Y DIAGNÓSTICO	31

4.1.1 Investigación previa	31
4.1.2 Diagnóstico del sistema en el campo.....	31
4.2 DISEÑO DEL SISTEMA	32
4.2.1 Diseño del nuevo sistema de caracterización de parámetros climáticos para mini-jardines clonales	32
4.2.2 Diseño del módulo de control de encendido y apagado automático de luces para implementación de estrategias de control de luminosidad adecuadas para la operación del mini jardín- clonal.....	32
4.2.3 Análisis de deficiencias de los sensores actuales	33
4.3 DESARROLLO E IMPLEMENTACIÓN	33
4.3.1 Búsqueda de los sensores	33
4.3.2 Implementación del módulo de control de encendido y apagado automático de luces.	33
4.3.3 Uso de la Beaglebone con Ubuntu e implementación de scripts para el uso del modem USB.....	34
4.3.4 Optimización de las rutinas implementadas en el nodo sumidero o base.....	34
4.3.5 Establecimiento de comunicación entre Beaglebone y nodo base.	34
4.4 VERIFICACIÓN DE LA SOLUCIÓN.....	35
CAPÍTULO 5: DESCRIPCIÓN DETALLADA DE LA SOLUCIÓN	36
5.1 DESCRIPCIÓN GENERAL	36
5.2 DESCRIPCIÓN DEL HARDWARE	37
5.2.1 Nodo FFD (nodo base), Nodos RFD 1 y 2, y protocolo Miwi.....	37
5.2.2 Beaglebone y módem USB.....	38
5.2.3 Comunicación por protocolo I2C.....	40
5.2.4 Módulo de control de encendido y apagado automático de luces.	41
5.2 DESCRIPCIÓN DEL SOFTWARE	43
5.2.1 Rutinas de software implementadas sobre SIWA-RTOS	43
5.2.2 Rutinas de software de los nodos RFD.....	45
5.2.3 Rutinas de software del nodo FFD o nodo base.....	46
5.2.4 Ubuntu y librerías necesarias para la implementación de la solución.....	49
5.2.5 Scripts utilizados en la Beaglebone para conectarse a internet por medio del módem USB y para conectarse con el servidor de electrónica.....	50
5.2.6 Generación de archivos .csv.....	51
5.2.7 Software implementado en la Beaglebone y en el PIC32MX795F512L para establecer la comunicación por medio del protocolo I2C.....	55
5.2.8 Actualización de fecha y hora	57

5.2.9 <i>Ejecución programada de las tareas</i>	58
5.2.10 <i>Rutinas de software implementadas en la MMB</i>	58
CAPÍTULO 6: ANÁLISIS DE RESULTADOS	61
6.1 DIAGNOSTICO DEL SISTEMA A MEJORAR REALIZADO AL INICIO DEL PROYECTO	61
6.2 EVALUACIÓN DE LAS CONDICIONES OPERATIVAS ACTUALES Y EL USO DE TECNOLOGÍA MÁS ROBUSTA PARA APLICACIONES DE MONITORIZACIÓN DE AMBIENTES EXTREMOS PARA MEJORA DE LA VIDA ÚTIL DE LOS SENSORES	62
6.3 RED DE SENSORES: NODOS FFD Y RFD (PROTOCOLO MIWI)	64
6.3.1 <i>Mediciones de corriente en Nodo RFD</i>	65
6.4 USO DE LA BEAGLEBONE Y MODEM USB	69
6.5 RESULTADOS DE LOS SCRIPTS DE SUBIDA Y PARA COMUNICACIÓN I2C USADOS EN LA BEAGLEBONE	69
6.6 IMPLEMENTACIÓN DEL MÓDULO DE CONTROL DE ENCENDIDO Y APAGADO AUTOMÁTICO DE LUCES.....	72
CAPÍTULO 7: CONCLUSIONES Y RECOMENDACIONES	74
7.1 CONCLUSIONES	74
7.2 RECOMENDACIONES.....	75
BIBLIOGRAFÍA Y REFERENCIAS	76
APÉNDICES	79
A.1 GLOSARIO, ABREVIATURA Y SIMBOLOGÍA	79
A.2 ARCHIVOS GPRS Y CHATGPRS1	80

Índice de figuras

FIGURA 1. FUNDAMENTOS DE REDES DE SENSORES [19]	19
FIGURA 2. TARJETA DE EXPANSIÓN PARA STARTER KIT [20]	23
FIGURA 3. DIAGRAMA DE BLOQUES DE LA CAPAS DE COMUNICACIÓN INALÁMBRICA DE MICROCHIP (MIWI). [21]	24
FIGURA 4. ESQUEMA DE CONEXIÓN A INTERNET	26
FIGURA 5. ESQUEMA DE COMUNICACIÓN I2C.....	28
FIGURA 6. TRANSFERENCIA COMPLETA I2C [12]	29
FIGURA 7. TARJETA DE DESARROLLO MULTIMEDIA BOARD PARA PIC32MX. [22].....	30
FIGURA 8. SOLUCIÓN IMPLEMENTADA.....	37
FIGURA 9. MÓDEM USB HUAWEI E173 CONECTADO CON TARJETA BEAGLEBONE.....	39
FIGURA 10. CONEXIÓN PARA COMUNICACIÓN I2C	40
FIGURA 11. CONTROL DEL ENCENDIDO Y APAGADO DE LUCES POR MEDIO DE RELÉS	41
FIGURA 12. CONEXIÓN DE LA MMB CON LA ETAPA DE POTENCIA.....	42
FIGURA 13. ETAPA DE POTENCIA PARA CONTROL DE LOS RELÉS.....	43
FIGURA 14. MAIN DEL SOFTWARE DE LOS NODOS.....	44
FIGURA 15. DIAGRAMA DE FLUJO DE TAREA TASKMIWI EN NODOS RFD. [1]	45
FIGURA 16. DIAGRAMA DE FLUJO DE LA TAREA TASKMIWI EN NODO FFD	47
FIGURA 17. DIRECTORIO /HOME DE LA BB CON LOS SCRIPTS IMPLEMENTADOS	50
FIGURA 18. MÓDEM USB CONECTADO Y RECONOCIDO POR EL SISTEMA OPERATIVO	51
FIGURA 19. DIAGRAMA DE FLUJO DE SCRIPT UPLOADFILE.SH	52
FIGURA 20. DIAGRAMA DE FLUJO DE SCRIPT SFTP.SH	53
FIGURA 21. DIAGRAMA DE FLUJO DE SCRIPT PICI2C.SH	54
FIGURA 22. MÓDULO I2C HABILITADOS EN LA BEAGLEBONE	55
FIGURA 23. RUTINA PARA COMUNICACIÓN POR I2C IMPLEMENTADA EN EL PIC32 DEL NODO SUMIDERO.....	56
FIGURA 24. DIAGRAMA DE FLUJO DE SCRIPT UPDATE TIME.SH	57
FIGURA 25. ARCHIVO CRONTAB	58
FIGURA 26. VENTANAS DISEÑADAS MEDIANTE LA HERRAMIENTA GRAPHICS DISPLAY DESIGNER.....	59
FIGURA 27. PRIMER ENSAYO PARA MEDICIÓN DE CORRIENTE EN NODO FFD	65
FIGURA 28. SEGUNDO ENSAYO PARA MEDICIÓN DE CORRIENTE EN NODO FFD	66
FIGURA 29. TERCER ENSAYO PARA MEDICIÓN DE CORRIENTE EN NODO FFD	67
FIGURA 30. PRUEBA DE SUBIDA DE ARCHIVOS AL SERVIDOR DE ELECTRÓNICA	70
FIGURA 31. ARCHIVO FECHA-DATA.CSV SUBIDO AL SERVIDOR	71
FIGURA 32. ARCHIVO FECHA-LOG.CSV SUBIDO AL SERVIDOR	71
FIGURA 33. SISTEMA DE CONTROL DE ENCENDIDO Y APAGADO AUTOMÁTICO MONTADO Y FUNCIONANDO EN EL INVERNADERO ..	73
FIGURA 34. ILUMINACIÓN MONTADA Y FUNCIONANDO EN EL INVERNADERO.....	73

Índice de tablas

TABLA 1. CARACTERÍSTICAS DE LA BEAGLEBONE [18]	27
TABLA 2. ESPECIFICACIONES TÉCNICAS DE LA TARJETA DE DESARROLLO MULTIMEDIA BOARD PARA PIC32MX	30
TABLA 3. CÓDIGO DEL INDICADOR LUMINOSO DEL MÓDEM USB HUAWEI E173	39
TABLA 4. COMANDOS EMPLEADOS EN LA COMUNICACIÓN ENTRE LA BEAGLEBONE Y EL PICMX795F512L	48
TABLA 5. RESULTADOS DE EVALUACIÓN DE DIAGNÓSTICO INICIAL DEL SISTEMA	61

Capítulo 1: Introducción

En la actualidad la protección ambiental es primordial, y en Costa Rica los esfuerzos por tener una actitud conservacionista son importantes. Nuestro país posee gran parte de su territorio dedicado a la conservación del medio ambiente, alrededor de unos 13.000 kilómetros cuadrados en parques y reservas nacionales (25% del territorio).

La Escuela de Ingeniería en Electrónica del Instituto Tecnológico de Costa Rica ha puesto en marcha un proyecto de investigación para diseñar una red de sensores capaz de cumplir con las labores de monitoreo que se requieran realizar en las zonas de protección ambiental. Este proyecto se llama CRTECMote y es una plataforma de hardware y software, que permite aprovechar los recursos de hardware y contar con un bajo consumo energético, así como también hacer un uso eficiente del tiempo disponible, usando un RTOS³.

Por otro lado, desde el año 2001 viene funcionando la primera cooperativa de mejoramiento forestal en Costa Rica, denominada GENFORES, la cual tiene como misión promover la exploración, conservación, utilización racional y mejoramiento de los recursos genéticos forestales. La Escuela de Ingeniería Forestal del Instituto Tecnológico, es el ente científico y director de la cooperativa.

Actualmente el TEC con sede en Santa Clara de San Carlos, tiene en sus instalaciones sistemas de producción en invernadero, donde sus propios ingenieros ya producen clones de árboles a gran escala para plantar, especies tales como la teca, melina, y un grupo de especies nativas. La mecánica seguida es que cuando una empresa entra a GENFORES, la misma compañía asigna un ingeniero con su respectivo grupo de trabajadores, quienes serán los responsables del desarrollo de su propio programa de mejoramiento genético, para lo cual se inicia una capacitación intensiva sobre clonación en las instalaciones de GENFORES.

³ RTOS: Sistema operativo de tiempo real.

Es por ello, que con el fin de contribuir a la gran labor del equipo de GENFORES y servir como plataforma de pruebas para la red inalámbrica de sensores CRTECMote desarrollada por la Escuela de Ingeniería Electrónica, se implementó el proyecto de caracterización de parámetros climáticos en mini jardines clonales forestales bajo ambiente protegido en su primera etapa por el ingeniero Jose Pablo Fallas Zúñiga. [1]

1.1 Problema existente e importancia de su solución

1.1.1 Definición del problema

En este lugar se cuenta con una serie de túneles con ambiente controlado, o mini jardines clonales para la reproducción de los clones de árboles creados por ellos. Este ambiente controlado debe presentar características climatológicas adecuadas para el óptimo desarrollo de los árboles en su etapa inicial tales como humedad relativa, temperatura y cantidad de luz, entre otras.

Algunas de estas condiciones son manipuladas mediante sistemas de riego por micro-aspersión, que se realiza de manera automática con temporizadores y sistemas de apertura de válvulas, además las decisiones de la frecuencia de riego son hechas a partir de la experiencia de los encargados y no de datos científicos o históricos.

A pesar de que el desempeño logrado en las recámaras con ambientes controlados es satisfactorio ante los requerimientos de GENFORES, no se cuenta con medidas de las características ambientales que permitan cuantificar su efecto en el desarrollo de los árboles. Ante esta situación, los encargados de GENFORES acudieron al personal de la Escuela de Ingeniería Electrónica, específicamente al Ing. Johan Carvajal Godínez, coordinador del proyecto de investigación CRTECMOTE para analizar la posibilidad de implementar un sistema que permita dar solución a esa problemática.

Este sistema fue implementado en una primera etapa en su proyecto de graduación por el entonces estudiante Jose Pablo Fallas Zúñiga. El sistema funcionó correctamente para los objetivos planteados. Sin embargo, la solución empleada requiere de mejoras ya que el sistema de red de sensores implementado funciona intermitentemente debido a sobrecarga en el nodo sumidero. Este nodo se encarga de recopilar la información proveniente de los otros nodos los cuales tienen una conexión analógica y por el protocolo I²C con los sensores. Este nodo sumidero además de esto, se encarga de la comunicación vía Ethernet con un punto de acceso, y se encarga de mostrar toda la información mediante una aplicación web. Debido que este nodo funciona como servidor de esta aplicación entre sus otras labores, presenta un recarga de trabajo y por lo tanto se sobrecalienta, o en otras palabras, la eficiencia no es la adecuada, lo que no permite su funcionamiento continuo por varias horas.

Por otro lado, los sensores escogidos no cumplen satisfactoriamente con los requerimientos de las mediciones en cuanto a confiabilidad debido a sus continuas fallas. Además, su empleo en ambientes exteriores provoca una reducción en su vida útil, principalmente cuando se presentan condiciones extremas como altas temperaturas con una humedad relativa muy elevada. Por esta razón es necesario sustituirlos constantemente por mal funcionamiento, lo cual no es lo más adecuado desde el punto de vista económico.

También se requiere de un sistema de iluminación el cual permita extender el fotoperiodo de las plantas como parte de los experimentos de GENFORES, este sistema debe encenderse y apagarse automáticamente a determinadas horas preestablecidas anteriormente. El sistema de caracterización de parámetros viene a cuantificar el efecto de estas acciones. Por ello es necesaria la implementación de mejoras en el funcionamiento del nodo sumidero.

1.1.2 Síntesis del problema

El sistema de monitorización de mini-jardines clonales implementado en las instalaciones de GENFORES en la sede del TEC de Santa Clara por un lado funciona intermitentemente debido a fallas por sobrecarga del nodo sumidero y por otro lado hay poca confiabilidad en los sensores utilizados.

1.1.3 Importancia de la solución

Este proyecto encargado de la caracterización de parámetros climáticos es de suma importancia por su contribución con la labor del equipo de GENFORES en la búsqueda del mejoramiento de los recursos genéticos forestales, ya que permite mediante datos cuantitativos caracterizar las condiciones más óptimas para el desarrollo de los árboles en su etapa inicial, tales como humedad relativa, temperatura y cantidad de luz.

De igual forma, la puesta en marcha de esta solución sirve como plataforma de pruebas para la red inalámbrica de sensores CRTECMote. Se pondrá contar con un sistema que permite comunicar la información desde el lugar donde se encuentran los nodos con sus sensores, hasta cualquier punto donde se tenga acceso a internet. Esto permite extender las aplicaciones de los nodos CRTECMote bajo una red inalámbrica, donde se requiera de otros tipos de transferencia y manejo de la información recolectada.

1.2 Solución seleccionada

Tomando en cuenta que el propósito principal de la solución al problema planteado es darle un funcionamiento óptimo en primer lugar al nodo sumidero o receptor, para que la red funcione correctamente, sin perder de perspectiva que la importancia de ello es poder tener acceso desde cualquier punto a la información recolectada por los sensores, se deben realizar cambios de software y de hardware también. Para poder acceder a la información recolectada desde internet y no solo desde un área local, se requiere del cambio en la arquitectura. Para

brindar esa solución en primer lugar se debe optimizar el software implementado en el nodo receptor para disminuir su consumo energético, para lo que se considera es apropiado eliminar todas las funciones encargadas del manejo del protocolo TCP/IP en primera instancia, en otras palabras, eliminar la comunicación vía Ethernet entre este nodo y un enrutador inalámbrico que permitía la conexión en una red de área local, porque ya no se hace necesaria con las modificaciones en la arquitectura, y porque el ineficiente empleo de esta tarea es la principal causante del sobrecalentamiento.

Por lo que la solución seleccionada será la implementación de un dispositivo que pueda encargarse de subir a internet la información recolectada por el nodo sumidero, obtenida anteriormente de los diferentes nodos conectados con los sensores. Para ello deberá utilizarse una tarjeta de desarrollo PIC 32 Starter Kit como nodo sumidero la cual se comunique por protocolo i^2c con una tarjeta Beaglebone la cual a su vez está conectada con una datacard, para subir la información a internet, sustituyendo el enrutador inalámbrico y la comunicación vía Ethernet.

Como hardware para implementar el encendido y apagado automatizado de luces para extender el fotoperiodo de algunas plantas, se seleccionó la tarjeta de desarrollo Multimedia Board para PIC32MX. Este dispositivo permite implementar el control necesario y brinda facilidades de interacción con el usuario por medio de su pantalla táctil.

Capítulo 2: Meta y Objetivos

2.1 Meta

Lograr que el proyecto de caracterización de parámetros climáticos en mini jardines clonales forestales bajo ambiente protegido alcance un funcionamiento óptimo y continuo para acceder a su información desde cualquier punto dentro de la red institucional del ITCR.

2.2 Objetivo general

Optimizar el diseño e implementación de la red inalámbrica de sensores utilizada en la monitorización y control de los mini-jardines clonales mediante un análisis de la operación del sistema actual con el propósito de un incremento en la confiabilidad y accesibilidad de los datos recolectados.

2.3 Objetivos específicos

- Desarrollar una nueva arquitectura de red que permita utilizar el hardware actual de la red de sensores y otros componentes adicionales para brindar una accesibilidad de los datos vía Internet con una disponibilidad completa.
- Incrementar al doble de años la vida útil de los sensores mediante la evaluación de las condiciones operativas actuales y el uso de tecnología más robusta para aplicaciones de monitorización de ambientes extremos.
- Integrar un módulo de control de encendido y apagado automático de luces que permita implementar estrategias de control de luminosidad adecuadas para la operación del mini jardín clonal.

Capítulo 3: Marco Teórico

3.1 Descripción general del sistema a mejorar

Con el fin de caracterizar los parámetros climáticos en los mini-jardines clonales se implementó en un invernadero ubicado en el Instituto Tecnológico de Cartago con sede en Santa Clara de San Carlos el sistema electrónico correspondiente para ese fin.

Este sistema consta de una red inalámbrica de sensores bajo la arquitectura CRTECMote la cual se encarga de tomar medidas de parámetros climáticos como humedad relativa, temperatura y luz. Los datos recopilados por la red de sensores son visibles desde cualquier navegador web pero en un área local.

Esta red implementada consta de tres nodos ubicados en distintos puntos del invernadero los cuales tiene conectados los sensores respectivos para la recolección de datos. Se cuenta con un nodo sumidero o nodo base el cual recibe la información de proveniente de estos tres nodos por medio del protocolo de comunicación inalámbrica Miwi. Este nodo base además cuenta con rutinas para el manejo del protocolo TCP/IP para que a través de una conexión Ethernet y un enrutador, se pueda visualizar la información de los sensores desde un navegador web en una red de área local. Para el manejo de la base de datos, el nodo base también se encarga de manejar rutinas USB para guardar y escribir en una tarjeta de memoria.

El hardware empleado fueron los sensores de luz BPW34, los de temperatura y humedad relativa HYT-271, las cuatro tarjetas de desarrollo Starter Kit con sus respectivas tarjetas de expansión para realizar las conexiones de los sensores (1 tarjeta para nodo base y 3 para los demás nodos), el enrutador y todos los materiales necesarios para el montaje y encapsulación de un sistema electrónico de esta clase en un invernadero.

3.2 Antecedentes bibliográficos

En la investigación realizada, cabe mencionar que en este proyecto se busca la optimización del sistema de caracterización de parámetros en mini-jardines clonales implementado por el Ing. Jose Pablo Fallas, por lo que se toma como importante base bibliográfica el informe presentado para ese proyecto.

Por otro lado, cabe mencionar que un gran aporte bibliográfico en el desarrollo e implementación de scripts para un sistema operativo como Linux en este proyecto, y una gran ayuda fueron los innumerables foros y blogs de comunidades dedicadas al apoyo del desarrollo de software libre.

Además antecedentes bibliográficos en el uso de bibliotecas de microchip son también un gran aporte, ya que aunque son para aplicaciones generalizadas, su documentación permite adecuarlas a las necesidades propias de un proyecto específico.

3.3 Descripción de los principales principios físicos y/o electrónicos relacionados con la solución del problema

3.3.1 Redes de sensores

“Esta tecnología emergente de redes inalámbricas de sensores ofrece una amplia gama de posibilidades para la implementación de aplicaciones que operen en ambientes con restricción en la cantidad de energía disponible, o ambientes de difícil acceso. Se puede afirmar que una red inalámbrica de sensores es un sistema de procesamiento distribuido.” [2]

El desarrollo de estas redes de sensores va de la mano con los avances en materia de comunicación inalámbrica, la cual requiere del empleo de técnicas para alcanzar una comunicación efectiva y confiable reduciendo el costo de su implementación, bajando el consumo energético al mínimo posible, reduciendo las dimensiones de los nodos, entre otros retos. [3]

Cuando se habla de redes de sensores se deben tomar en cuenta importantes características como la topología empleada, la tolerancia a errores, el consumo energético, las limitaciones de hardware, la variabilidad del canal, comunicaciones multisalto, etc., y considerando estas variables evaluar su aplicación en campos como eficiencia energética, sensores ambientales e industriales, medicina, domótica, entornos de alta seguridad y demás campos que lo puedan emplear.

En la figura 1 se muestran fundamentos importantes de las redes de sensores, como la arquitectura que consta de los diversos nodos (motes), los nodos base, el punto desde donde se accesa a la información recolectada. La topología empleada puede tener diferentes configuraciones como la estrella o la malla entre otras. La comunicación puede utilizar diferentes tecnologías inalámbricas:

- Zigbee (IEEE 802.15.4)
- UWB (Ultra Wide Band)
- Bluetooth (IEEE 802.15.1)
- RFID (Radio Frequency Identification)
- WiFi (IEEE 802.11)
- ...



Figura 1. Fundamentos de redes de sensores [19]

3.3.2 Sensores

Para este proyecto las características climatológicas tomadas en cuenta en los cultivos bajo ambientes controlados son la cantidad de luz (medida en Lux), la humedad relativa del aire y la temperatura del ambiente. Por esta razón se mencionan algunos aspectos importantes en las mediciones de cada factor.

3.3.3 Sensores de temperatura

La temperatura atmosférica es el indicador de la cantidad de energía calorífica acumulada en el aire. Los instrumentos de medición de temperatura se pueden basar en diversos principios físicos, los métodos mecánicos son la variación del volumen, o variaciones de presión a volumen constante. Los métodos eléctricos se basan en variaciones de resistencia de un conductor (Sondas de resistencia), variación de resistencia de un semiconductor (termistores), diferencia de potencial creada a partir de la unión de dos metales (termopares), basados en la radiación (pirómetros de radiación). [4]

3.3.4 Sensores de Humedad Relativa

“La humedad relativa dada en los informes meteorológicos, es la razón entre el contenido efectivo de vapor en la atmósfera y la cantidad de vapor que saturaría el aire a la misma temperatura”. [5]

“Existen diversos métodos usados para medir el porcentaje de humedad relativa, uno de ellos se basa en la propiedad de algunos materiales de presentar diferente elasticidad con la humedad, estos son los higrómetros de cabello”. Sin embargo su uso en invernaderos no es adecuado por sus limitaciones en el rango de medida. Los psicrómetros se basan en las medidas de las temperaturas de un termómetro seco y un termómetro húmedo. El primero determina la temperatura del ambiente y el segundo marca una temperatura diferente a la anterior, ya que

va en función de la cantidad de agua evaporada y, por tanto, de la presión relativa del vapor de agua en la atmósfera. La desventaja es que hay que mantener permanentemente húmeda la mecha del segundo termómetro”. [6]

Los sensores capacitivos son los más recomendados actualmente por sus grandes ventajas, fundamentalmente porque se pueden conectar fácilmente a equipos de control automático, sin embargo el principal inconveniente es que a humedades altas el dieléctrico se satura (100% HR). Por otro lado, se encuentran también los higrómetros ópticos de punto de rocío, el cual es considerado el método más preciso de medición. La precisión en la medida es muy alta, pero su desventaja es su elevado coste versus la necesidad de precisión en la medición. [6]

3.3.5 Sensores de luz.

“Para la detección y medición de luz existen una gran cantidad de dispositivos como los piranómetros, sensores quantum y luxómetros, todos ellos de gran complejidad. Sin embargo, su funcionamiento se basa en dispositivos de menor complejidad conocidos como detectores fotovoltaicos tales como los fotodiodos, los fototransistores y las fotorresistencias. Cualquiera de los dispositivos foto detectores puede ser utilizado para la medición de luz en cualquier unidad deseada siempre y cuando se realicen los acondicionamientos de señal necesarios”. [1]

Por lo tanto existe gran cantidad de sensores de luz disponibles en pequeños encapsulados a precios razonables. En la mayoría de ocasiones la mejor solución para contar con un sensor de luz es integrar un fototransistor o un fotodiodo con un amplificador de corriente. [7]

3.3.6 PIC 32 Starter kit

El PIC32 Starter kit es uno de los productos de la empresa Microchip, el cual es un kit de desarrollo y una herramienta apta para el diseño de sistemas

embebidos de bajo costo y con gran versatilidad. Este dispositivo cuenta con un microcontrolador de 32 bits, y con conectividad USB para su misma programación y depuración, además cuenta con tres botones y tres LED's para disposición del usuario. Esta tarjeta puede traer una plataforma para desarrollar aplicaciones Ethernet o USB, o puede no traerlas, esto varía según el modelo. Además pueden usar diversos microcontroladores de la familia PIC32MX.

Uno de los microcontroladores más completos con los que cuentan estas tarjetas es el PIC32MX795F512L el cual puede trabajar a una frecuencia máxima de 80MHz, su arquitectura es MIPS32 M4K con 5 estados de pipeline, opera en un rango de voltajes entre 2.3V y 3.6V, cuenta con una memoria Flash de 512K, memoria SRAM de 128K. En cuanto a sus periféricos, puede trabajar con USB 2.0, Ethernet, posee 6 módulos UART, 4 módulos SPI, 5 módulos para comunicarse por protocolo I2C, cuenta con RTCC, 5 temporizadores, 16 pines para entradas analógicas, entre otros.

Otros de los microcontroladores usados son el PIC32MX360F512L y el PIC32MX460F512L los cuales cuentan básicamente con la misma arquitectura del microcontrolador mencionado anteriormente, solo que con algunas diferencias en cuanto a capacidad de sus memorias, y cantidad de módulos periféricos, ya que en estos modelos es más reducida la capacidad, pero si cuentan al menos con los módulos mencionados en el párrafo anterior.

Otro punto importante a tocar con respecto a estas tarjetas de desarrollo, es la conectividad con otros dispositivos, ya que tal y como se observa en la figura 2, no se tiene acceso directo a sus puertos, para ello, es necesario acoplarle una tarjeta adicional llamada "Starter Kit I/O Expansion Board", la cual provee un acceso completo a la señales de MCU. Con la tarjeta de expansión, es posible conectarle al PIC32, dispositivos como transceptores, pantallas LCD, establecer comunicación con otros microcontroladores, etc.

En la figura 2 se puede apreciar como está constituida la tarjeta de expansión.

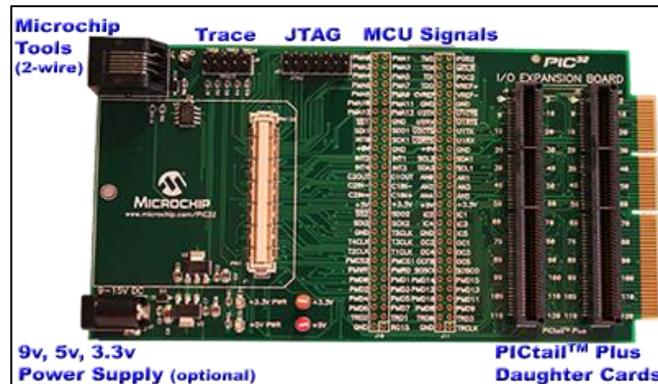


Figura 2. Tarjeta de expansión para Starter Kit [20]

3.3.7 Miwi

Miwi es un protocolo inalámbrico para redes de área personal con tasas bajas de transmisión de datos, basado en el estándar IEEE 802.15.4. Entendiendo por red de área personal la comunicación entre distintos dispositivos cercanos al punto de acceso, normalmente de pocos metros y para uso personal.

Este protocolo está dirigido a redes que no necesitan altas transferencias de datos (250Kbit/s), corta distancia (100m sin obstáculos), bajo consumo energético y bajo costo. Fue desarrollado por Microchip Technology, como un proyecto de código abierto, se encuentra dentro del espectro de los 2.4GHz a través de su transceptor MRF24J40. Con respecto a las topologías de red, este protocolo puede formar redes punto a punto, de estrella o malla. En cuanto a capacidad máxima de infraestructura, Miwi puede tener 8 coordinadores como máximo por red, y cada uno de éstos 127 hijos, haciendo un total de 1024 nodos por red. [8]

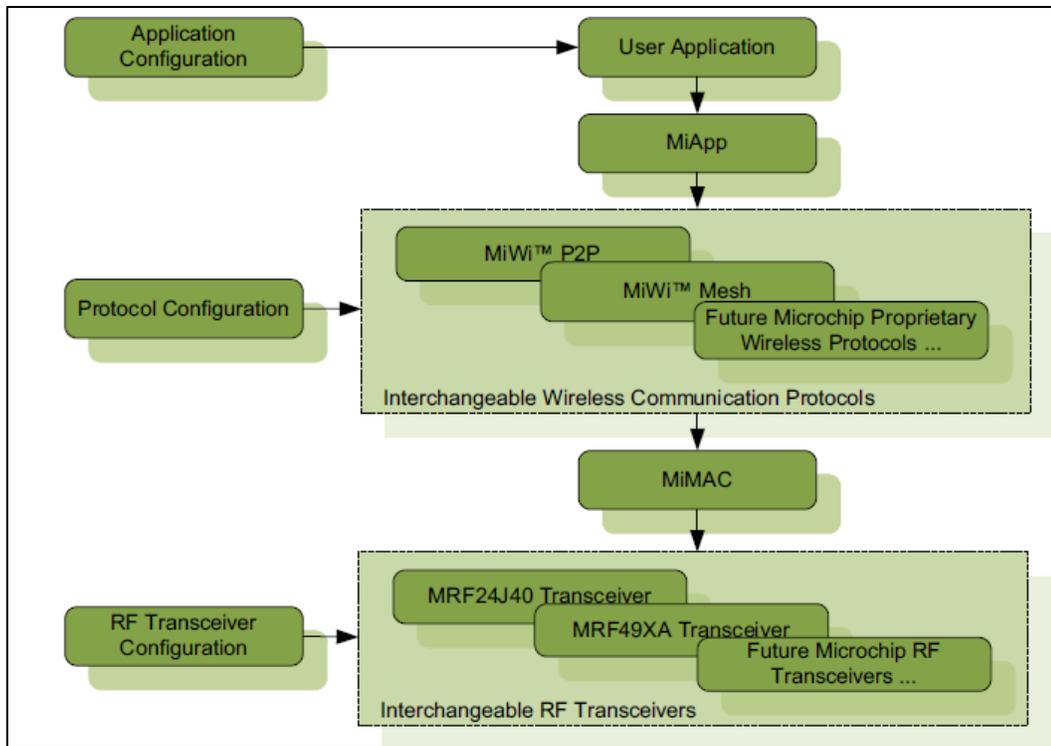


Figura 3. Diagrama de bloques de la capas de comunicación inalámbrica de microchip (Miwifi). [21]

En la figura 3, las diferentes capas del protocolo de comunicación, para las cuales microchip dispone de las librerías necesarias para su implementación en el microcontrolador correspondiente, y que se encuentran en el lenguaje de programación C.

3.3.8 Modem USB

Los modem USB también conocidos como “datacard” o modem USB 3g, son dispositivos electrónicos que permiten mediante una línea de telefonía móvil conectar equipos portátiles a internet. Estos dispositivos utilizan una tarjeta SIM (integrada o removible) para identificarse ante la tele operadora.

“Los términos *modem 3G* y *router 3G* son nombres genéricos que se utilizan para designar aquellos equipos electrónicos que permiten conectar con una red de telefonía móvil (GPRS, UMTS, HSPA, EVDO, 3G) para conseguir acceder a

Internet desde un dispositivo portátil o de sobremesa. Estrictamente hablando, los modem y los router se diferencian en que los modem sólo dan servicio a un único equipo, mientras que los router permiten que todos los equipos de una red local tengan acceso a Internet al mismo tiempo de forma transparente.” [9]

La implementación de este tipo de dispositivos posee gran cantidad de beneficios como que permite acceso a internet desde cualquier punto donde se cuente con cobertura celular, además gracias a su puerto USB pueden ser utilizados con gran cantidad de dispositivos, en muchos casos la velocidad de conexión es similar al ADSL convencional, entre otros. Sin embargo también se cuenta con desventajas tales como que la velocidad de conexión y estabilidad depende de su localización y cobertura, ciertas aplicaciones están limitadas como los programas de intercambio P2P.

En Costa Rica desde diciembre del 2009 se pusieron a disposición de los usuarios las líneas móviles de tercera generación (3G). En nuestro país empresas como Kolbi, Movistar, Claro y TUYO Móvil distribuyen este tipo de dispositivos.

3.3.9 PPP (Protocolo punto a punto)

Para lograr conectarse a internet, una de las formas más comunes para un usuario es a través de una línea telefónica o un módem. Para lograr esto, se requiere de un protocolo punto a punto de enlace de datos en la línea, para el manejo de marcos de errores y demás funciones de la capa de enlace de datos, la cual es responsable de la transferencia fiable de información a través de un circuito de transmisión de datos, recibiendo peticiones de la capa de red y utilizando la capa física. Por lo tanto, este protocolo permite establecer comunicación entre dos computadoras a nivel de la capa de enlace TCP/IP, sin embargo cabe mencionar que dos protocolos de este nivel utilizados ampliamente en Internet son SLIP (Serial Line Internet Protocol) y PPP (Point to Point Protocol).

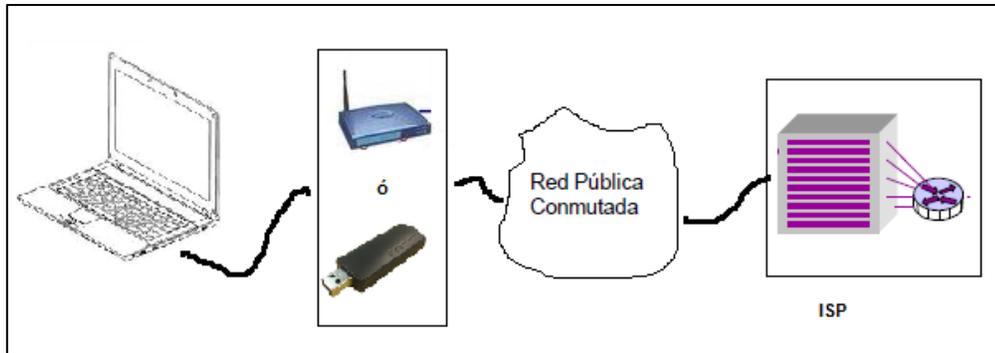


Figura 4. Esquema de conexión a internet

En la figura 4 se observa un esquema general de conexión a internet donde “muchos usuarios suelen acceder a redes interconectadas mediante TCP/IP: Un modem conectado por un enlace serie a un proveedor de servicio de internet (ISP). Entre el ISP y el ordenador conectado al modem se establece un enlace controlado por un protocolo especial (PPP).” [10]

Algunas características importantes por mencionar de este protocolo son que sus enlaces de comunicación son full-dúplex, es el más usado para acceder a líneas TCP/IP a través de líneas serie, poco a poco va sustituyendo al SLIP, y entre sus constituyentes se encuentra un mecanismo de encapsulación para enlaces IP punto a punto, un protocolo de control de enlace y una familia de protocolos para configurar los protocolos de red.

Sistemas operativos como Linux contienen librerías para el uso de estos protocolos los cuales permiten la conexión a internet a través de dispositivos como los modem USB. Estas librerías hacen uso de tecnologías como GPRS (servicio general de paquetes vía radio). “GPRS es una nueva tecnología que comparte el rango de frecuencias de la red GSM utilizando una transmisión de datos por medio de 'paquetes'. La conmutación de paquetes es un procedimiento más adecuado para transmitir datos, hasta ahora los datos se habían transmitido mediante conmutación de circuitos, procedimiento más adecuado para la transmisión de voz.” [11]

3.3.10 Beaglebone

La tarjeta Beaglebone es la última adición a la familia BeagleBoard.org y como sus predecesores, está dirigida a la comunidad desarrolladora de código abierto (Open Source Community), o a cualquier interesado en procesadores de bajo costo ARM Cortex A8. Esta tarjeta ha sido equipada con un set de características mínimas suficientes si para aprovechar el potencial del procesador.

Tabla 1. Características de la Beaglebone [18]

Especificaciones	Características	
Procesador	AM3359 500MHz-USB alimentado 720MHz-CD alimentado	
Memoria	256MB DDR2 400MHz (128MB opcional)	
PMIC TPS65217B	Reguladores de potencia	
Soporte para depuración	Adaptador USB-serial	Conector mini-USB
	USB vía JTAG	4 LED's para usuario
Alimentación	USB	5V CD externo
Puerto HS USB 2.0 client	acceso USB1 modo cliente	
Puerto HS USB 2.0 host	USB tipo A.	
Ethernet	10/100, RJ45	
SD/MMC conector	microSD, 3.3V	
Interfaz de usuario	1 botón de reset	
Protección sobre voltaje	5,6V máx. Se apaga si excede.	
Conectores de expansión	Alimentación 5V, 3.3V, SPI1, I2C, GPIO, LCD, MMC1, MMC2, 3 puertos seriales, 4 temporizadores, entre otros	
Peso	1,4 oz (39,68 g)	

Esta tarjeta es capaz de correr sistemas operativos como Armstrong o distribuciones de Linux como Ubuntu, los cuales se inician de una tarjeta microSD.

Su capacidad de manejar un sistema operativo como Ubuntu, el puerto USB con el que cuenta, el puerto para conexión vía Ethernet, su bajo costo y su tamaño compacto son algunos de los beneficios más importantes que brinda esta tarjeta.

3.3.11 Protocolo de comunicación I²C

“Para simplificar la interconexión de dispositivos al microprocesador, Philips desarrolló un sencillo bus bidireccional basado en dos hilos por el que se transmiten los datos vía serie y lo llamó El Bus I2C. EL Bus I2C (Inter- Integrated Circuits) fue desarrollado al principio de los 80’s. Su propósito original fue el de proporcionar una manera fácil de conectar un CPU a los chips periféricos en un equipo de TV.” [12]

Esta comunicación serial consta de dos líneas bidireccionales, donde una de ellas está encargada de manejar los pulsos de reloj (SCL) y la otra línea maneja el intercambio de datos (SDA), entre los dispositivos conectados al bus. Cabe mencionar que también es necesaria una tercera línea pero esta sólo es la referencia. Las líneas SDA y SCL deben conectarse al positivo de la alimentación a través de las resistencias de drenador abierto. Si el bus está libre las líneas se encuentran en alto. El dispositivo puede ser considerado maestro o esclavo. El dispositivo maestro es el que genera la señal de reloj y de igual manera es el que inicia la transferencia en el bus. Cada dispositivo conectado al bus tiene un código o una dirección única que puede ser seleccionada mediante software. En la figura 5 se muestra la conexión.

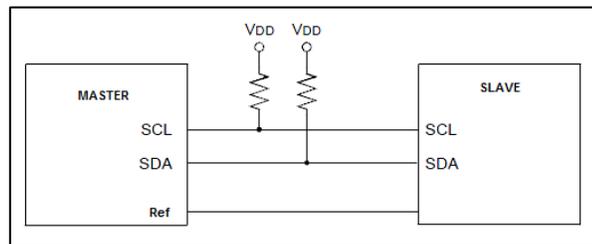


Figura 5. Esquema de comunicación I2C

En cuanto al formato del mensaje, primeramente el maestro inicia la comunicación generando una condición de inicio, luego se envía la dirección y la orden de escritura, el esclavo direccionado responde con señal de conformidad (ACK), luego el maestro envía los datos, y de nuevo el esclavo responde con ACK, y finalmente el maestro termina la comunicación enviando una condición de

parada. La comunicación en este protocolo suele ser con palabras de 8 bits, sin embargo existen algunos dispositivos, como micro-controladores los cuales son capaces de emplear este protocolo con palabras de 10 bits.

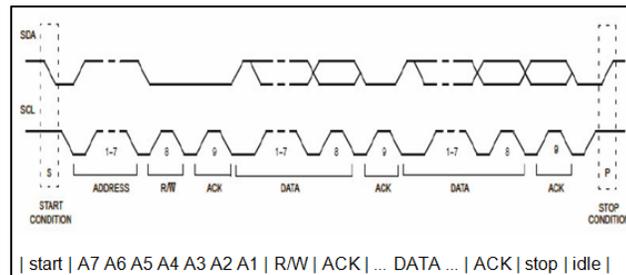


Figura 6. Transferencia completa I2C [12]

En la figura 6 se muestra según los ciclos de reloj como se da la transferencia de información mediante el protocolo I2C.

3.3.12 Multimedia Board PIC32MX

Es una tarjeta de desarrollo fabricada por la empresa Microchip. Esta tarjeta es ideal para la implementación del proyecto ya que contiene varios módulos necesarios para el sistema además de tener el microcontrolador acoplado. La siguiente lista detalla algunos de los elementos útiles para el proyecto que se encuentran ensamblados en la tarjeta:

- Microcontrolador PIC32MX460F512L32-bit
- Cristal de 8 MHz
- Pantalla táctil
- Puertos de entrada/salida que se pueden mapear en sus extremos

En la figura 7 se observa la MMB y sus diversas partes.

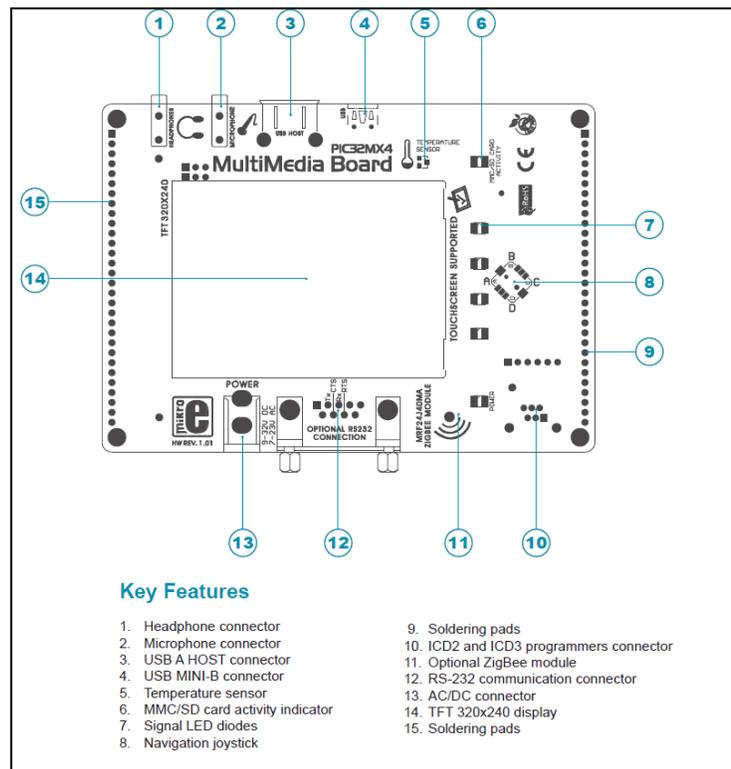


Figura 7. Tarjeta de desarrollo Multimedia Board para PIC32MX. [22]

Algunas de las especificaciones más importantes de esta tarjeta de desarrollo se resumen en la tabla 2.

Tabla 2. Especificaciones técnicas de la tarjeta de desarrollo Multimedia Board para PIC32MX

Especificaciones	Característica
Procesador	PIC32MX460F512L 32-bit
Fuente de alimentación eléctrica	Conexión por USB Alimentación CD 5V
Memoria Flash adicional	8Mbit
Oscilador interno	8MHz
Pantalla táctil	Display TFT con resolución 320x240
Conector MMC/SD	Comunicación vía SPI

Capítulo 4: Procedimiento metodológico

En este capítulo se detallan los pasos seguidos para alcanzar la solución seleccionada. Las actividades realizadas se agruparon en cuatro grandes categorías, según su orden cronológico.

4.1 Investigación y diagnóstico

4.1.1 Investigación previa

Consulta bibliográfica propia de cualquier proyecto, para tener una visión clara del problema y encontrar un enfoque adecuado de la solución. Se debió investigar aspectos técnicos de SIWA (CRTECMote), en el entorno del proyecto de GENFORES, aspectos técnicos de comunicación inalámbrica tanto de MIWI como de comunicación punto a punto, protocolos de comunicación como SPI o I2C, disminución del consumo de potencia en la red de sensores, entre otros.

También fue necesario como parte de la investigación previa estudiar a fondo el proyecto implementado en primera instancia, para comprender todos los aspectos necesarios a tomar en cuenta, esto con el informe presentado de caracterización de parámetros para mini-jardines clonales.

4.1.2 Diagnóstico del sistema en el campo

Una vez comprendidas todas las funciones que el sistema montado podía realizar, y conociendo el hardware ubicado en el sitio, se realizaron visitas al ITCR con sede en Santa Clara, para llevar a cabo el diagnóstico del sistema.

Esto permitió identificar las condiciones de los dispositivos, ya que permitió determinar que se encontraba en un funcionamiento óptimo y cuales dispositivos habían dejado de funcionar. Sin embargo el diagnóstico también tomó en consideración aspectos propios del lugar y no específicamente del sistema implementado, como las condiciones de protección ante la lluvia, facilidades para

el uso de corriente eléctrica, prácticas habituales de los funcionarios del lugar, entre otros. Estas visitas de diagnóstico también permitieron palpar con una visión subjetiva las condiciones climáticas generales del lugar.

Resultados de este diagnóstico permitieron determinar que dispositivos era necesario reemplazar, y por lo tanto fue necesario entrar de nuevo en una etapa de investigación para la búsqueda por ejemplo de las características que deberían tener los nuevos sensores, adecuándose al lugar.

4.2 Diseño del sistema

4.2.1 Diseño del nuevo sistema de caracterización de parámetros climáticos para mini-jardines clonales

Habiendo analizado las deficiencias en la arquitectura de red implementada que son la causa del problema de sobrecarga en el nodo base, se diseñó la nueva solución, sustituyendo la comunicación vía Ethernet implementada, hardware como el enrutador inalámbrico, y la aplicación web usada, todo por el manejo de un modem USB controlado por la Beaglebone, la cual debe comunicarse con el nodo base, y brindar acceso a la información recopilada desde cualquier punto usando internet.

4.2.2 Diseño del módulo de control de encendido y apagado automático de luces para implementación de estrategias de control de luminosidad adecuadas para la operación del mini jardín-clonal

Uno de los objetivos específicos de este proyecto consta de añadir un módulo adicional al sistema de recolección de parámetros requerido para las investigaciones de GENFORES, por lo que se procedió a diseñar dicho sistema. Primero se determinaron los requerimientos de este módulo, número de elementos a controlar, hardware a implementar, tipo de interacción con el usuario entre otros. Para esto se tomaron decisiones como la implementación de la MMB como control principal de este módulo, la necesidad de añadir una etapa de acople de potencia,

y las consideraciones necesarias de la instalación de este sistema en los mini-jardines clonales.

4.2.3 Análisis de deficiencias de los sensores actuales

Búsqueda de los sensores adecuados y análisis de deficiencias de los actuales: es necesario buscar sensores aptos para su implementación en ambientes exteriores, los cuales sean confiables y además sean capaces de soportar las condiciones del lugar, teniendo en cuenta las deficiencias de los actuales, y los resultados de la evaluación realizada en el diagnóstico.

4.3 Desarrollo e implementación

4.3.1 Búsqueda de los sensores

Con base en el análisis y la búsqueda indicada en la sección 4.2.3, se señalan las características a tomar en cuenta en la nueva implementación de sensores y se recomienda el uso de determinados modelos, indicando la importancia en el acondicionamiento, el encapsulado y el montaje en el sitio.

4.3.2 Implementación del módulo de control de encendido y apagado automático de luces.

A partir del diseño inicial de este módulo, se implementa el sistema desarrollando las rutinas vía software para el control, para programar la MMB, se implementó el circuito impreso respectivo para el acople de potencia donde se encuentran los relés que encienden o apagan los focos o bombillos correspondientes. Con todo esto listo, se procedió a instalar el sistema en el invernadero de GENFORES, encapsulando el sistema descrito y posicionando los bombillos en los lugares requeridos.

4.3.3 Uso de la Beaglebone con Ubuntu e implementación de scripts para el uso del modem USB.

Con el objetivo claro de poder acceder a la información recolectada por los sensores desde internet, se procede a poner en funcionamiento un modem USB que utiliza tecnología 3g, el cual debe ser manejado por la Beaglebone. Para realizar esto, fue necesario en primera instancia levantar el sistema operativo de Linux, más específicamente su distribución Ubuntu. Para ello fue necesario realizar investigación adicional acerca del manejo de comandos de Linux, manejo de la tarjeta Beaglebone y como correr un sistema operativo como Ubuntu desde su tarjeta SD. Cuando se logró correr sobre esta plataforma, fue necesario buscar todas las librerías necesarias y crear los scripts necesarios para poner en funcionamiento el modem USB, y así poder subir información a internet.

4.3.4 Optimización de las rutinas implementadas en el nodo sumidero o base

En paralelo con el trabajo realizado con la puesta en funcionamiento del modem USB usando la Beaglebone, se procedió a optimizar la programación del nodo base, la cual emplea SIWA-RTOS en lenguaje de programación C, para la administración de las funciones del PICMX795F512L en la Starter kit. En este punto se eliminaron algunas funciones como la comunicación Ethernet usando el protocolo TCP/IP, se removieron funciones de comunicación vía USB con una tarjeta de memoria. En función de esta optimización también se añadieron rutinas de chequeo de conectividad entre los nodos.

4.3.5 Establecimiento de comunicación entre Beaglebone y nodo base.

Con las rutinas de software depuradas en los nodos, y con los scripts de la Beaglebone encargados del manejo del modem USB funcionando correctamente, era necesario establecer la comunicación entre ambos dispositivos. Para ello se evaluaron diversas alternativas de comunicación y se optó por el protocolo I2C, de modo que se establecieron parámetros en esa comunicación, como quien trabajaba como esclavo y quien como maestro, que tipo de información se iba

transportar, cuantos recursos era necesario usar y cuanto tiempo se iba disponer, entre otros.

4.4 Verificación de la solución

Para la verificación de la solución debieron realizarse las pruebas de campo del funcionamiento de este nodo base (junto con la red), por un tiempo prolongado en el lugar de implementación, para descartar problemas por sobrecarga. De igual forma se hicieron pruebas continuas de subida de archivos a través del módem USB, para establecer una estadística en la cantidad de errores a la hora de este proceso. También se evaluó diferencias en el proceso de subida ante variaciones en la intensidad de señal recibida por el módem.

Además, se realizaron mediciones en el consumo de corriente del nodo base, para determinar el nivel de optimización de las nuevas rutinas empleadas, para contrastar los resultados con el sistema sin optimizar. Además se pretende demostrar que se disminuyó la sobrecarga a este nodo.

Capítulo 5: Descripción detallada de la solución

Esta sección presenta el desarrollo de la solución propuesta al problema planteado, la cual describe el proceso realizado en dos grandes bloques, la descripción del hardware por un lado y en segundo plano la descripción del software. Sin embargo se comenta primeramente aspectos generales de la solución.

5.1 Descripción general

Desde el inicio del proyecto, se tiene claro que después de la primera implementación del sistema no se tiene acceso a los datos recopilados por los sensores a través de la aplicación web diseñada. A partir de aquí el supuesto problema podía encontrarse tanto desde la recopilación de los datos de los sensores, o en la transmisión a través de la red inalámbrica Miwi hasta el nodo base, o incluso en la comunicación vía Ethernet con el enrutador inalámbrico que finalmente permitía acceder a la información con la aplicación web mencionada.

Se realiza el diagnóstico y se determinan varios factores. Primeramente se encuentran daños en la totalidad de los sensores. Estos sensores se encontraban cubiertos de tierra, con agua en las conexiones, y por ende no estaban midiendo parámetros físicos. De los tres nodos instalados, dos se encontraban en óptimas condiciones, uno de ellos se encontraba dañado debido a que había agua en su circuitería. Se encontró además que el nodo base se encontraba funcionando bien, ya que efectivamente como era su función montaba la red Miwi, sin embargo era evidente su sobrecalentamiento al tacto. Además el enrutador inalámbrico se encontraba también dañado. También se comprobó el funcionamiento de la tarjeta de memoria conectada al nodo base, y no fue posible accederla, por lo que no era funcional.

Tomando en cuenta los resultados del diagnóstico, se desarrolla la solución propuesta con los siguientes cambios en el hardware y software.

5.2 Descripción del hardware

El esquema general del sistema de caracterización de parámetros anterior y la nueva solución implementada se muestran en la figura 8, donde se contrastan los cambios mostrando el nuevo hardware usado con el color rojo.

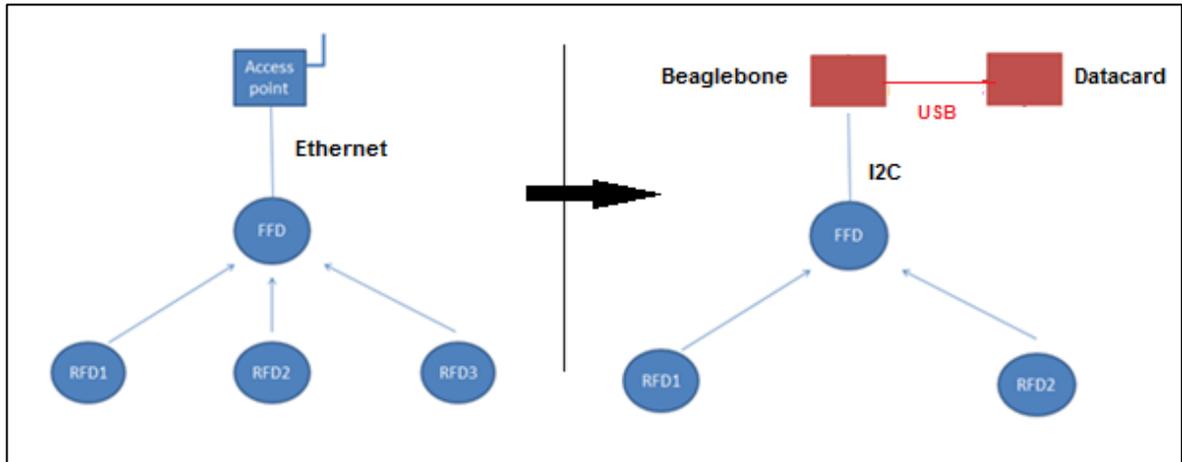


Figura 8. Solución implementada

En la figura 8 se observa como se sustituye el enrutador inalámbrico que se comunicaba por Ethernet con el microcontrolador del nodo base, y en su lugar se utiliza la tarjeta Beaglebone Rev A4, a la cual se le conecta el módem USB de la marca HUAWEI E173, que requiere de una línea de telefonía móvil para funcionar. Con el nuevo hardware, se emplea el protocolo de comunicación serial I2C, para la conexión entre el nodo base y la Beaglebone.

5.2.1 Nodo FFD (nodo base), Nodos RFD 1 y 2, y protocolo Miwi

Los nodos RFD son los dispositivos a los cuales se les conecta directamente los sensores, y se encuentran cerca del punto de medición. Para este proyecto se cuenta con dos nodos RFD.

Los nodos RFD se implementaron por medio de una tarjeta PIC32 Starter Kit, la cual se encuentra conectada a su tarjeta de expansión que permite conectarle el

transceptor. Estos nodos cuentan con un microcontrolador PICMX360F512L. El transceptor conectado a estos nodos se comunica con el MCU por medio de comunicación SPI.

De igual forma, para el nodo sumidero se emplea una Starter Kit con su respectiva tarjeta de expansión y su transceptor conectado, sin embargo el microcontrolador es diferente al de los nodos RFD, en este caso se utiliza un PICMX795F512L, el cual es un poco más capaz que los anteriores.

El protocolo usado en esta red de sensores es Miwi (Estándar IEEE 802.15.4), y para su utilización se utilizan los transceptores de la empresa microchip MRF24J40. Este se encuentra conectado al microcontrolador de los nodos por medio de la tarjeta de expansión. Su implementación es controlada por rutinas de software. Tanto en los nodos RFD como FFD, el transceptor se conectó con el módulo de SPI1, a pesar de que podía también utilizarse el módulo SPI2.

5.2.2 Beaglebone y módem USB

La tarjeta Beaglebone Rev A4 se convierte en la principal unidad de control de este sistema, ya que desde ella se controla no solo las peticiones de datos a la red de sensores, sino que también la subida de archivos a internet y funciona también como la base de datos, por lo que no es necesario la implementación de una tarjeta de memoria externa conectada al nodo base como fue implementada en el proyecto anterior. Esta tarjeta cuenta con el hardware necesario para comunicación por USB necesario para manejar la datacard, cuenta con el procesador suficiente para correr Ubuntu, cuenta con memoria no volátil y además contiene módulos para comunicación I2C.



Figura 9. Módem USB Huawei E173 conectado con tarjeta Beaglebone

Como se puede apreciar en la figura 9 esta tarjeta resulta muy cómoda debido a su pequeño tamaño, sus dimensiones son de 86.36mm x 53.34mm. Se alimenta con 5V en corriente directa, cuenta con un puerto Ethernet que aunque no se emplea en esta solución propuesta, si permite actualizarle paquetes de software a la hora de su programación. También a los costados de la tarjeta se encuentran pines que permiten la comunicación con sus puertos, que fueron usados para este proyecto para realizar la conexión por medio de I2C.

Con respecto al módem USB empleado, este funciona cuando se le añade una tarjeta SIM en su interior, que le permite hacer uso de la tecnología de telecomunicaciones. Se conecta por USB y cuenta con un indicador luminoso el cual sigue código de la tabla 3.

Tabla 3. Código del indicador luminoso del módem USB Huawei E173

Acción indicador	Significado
Verde: parpadea 2 veces cada 3s	MODEM USB encendido
Verde: parpadea 1 vez cada 3s	MODEM USB se está registrando en red 2G
Azul: parpadea 1 vez cada 3s	MODEM USB se está registrando en red 3G/3G+
verde constante	MODEM USB conectado a red 2G
azul constante	MODEM USB conectado a red 3G
Cian constante	MODEM USB conectado a red 3G+
Apagado	MODEM USB está desconectado

Es importante mencionar que se escogió utilizar un chip de la marca Kolbi, el cual utiliza la plataforma de telecomunicaciones del ICE en Costa Rica.

5.2.3 Comunicación por protocolo I2C

La conexión eléctrica de los dispositivos al bus I2C debe incluir resistencias de drenador abierto en cada una de las líneas SCL y SDA para asegurar que cuando no se está realizando ninguna transferencia se mantengan las líneas en alto ("1" lógico).

El valor de las resistencias usadas como drenador abierto para ambas líneas fue de 4.7k Ω , respetando la condición mínima establecida por la ecuación:

$$R_{P_{MIN}} = \frac{V_{DDMAX} - V_{OLMAX}}{I_{OL}} = \frac{3.3V - 0.4V}{6.6mA} = 439\Omega$$

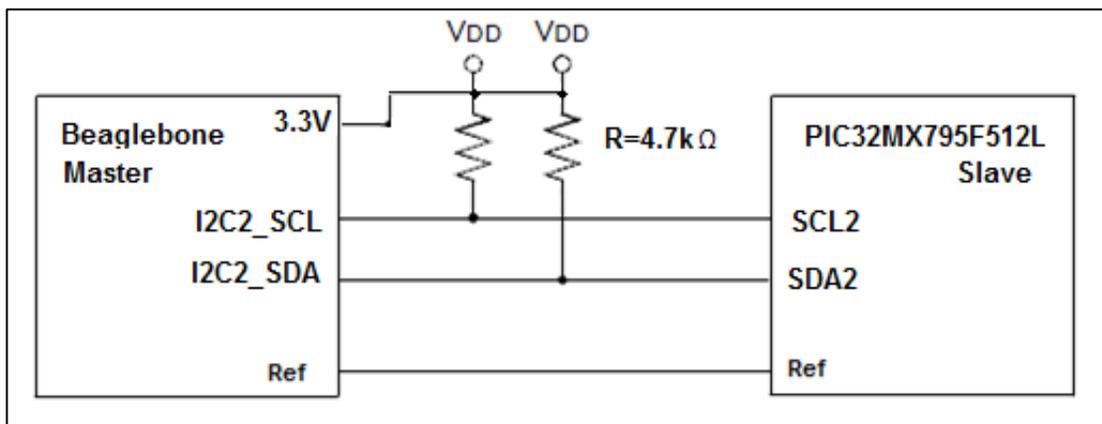


Figura 10. Conexión para comunicación I2C

En la figura 10 se muestra el esquemático de la conexión para el protocolo I2C, se debe mencionar que se utilizó la Beaglebone como dispositivo maestro y el microcontrolador del nodo FFD como dispositivo esclavo, de modo que quien controla la transmisión es la Beaglebone. Los pines utilizados para conectar la señal I2C2_SCL e I2C2_SDA fueron los pines 19 y 20 respectivamente del conector P9 en la tarjeta. Por otro lado, como se indica en la figura, en el MCU se conectaron las líneas de comunicación al módulo I2C2, las cuales se encuentran debidamente señaladas en la tarjeta de expansión.

La conexión del voltaje V_{DD} se tomó del pin 3 del conector P9 de la Beaglebone, así mismo la referencia se conectó a los pines 1 y 2 del mismo conector P9. En el dispositivo esclavo la referencia se conectó donde se encuentra indicado con las letras “GND”, en la tarjeta de expansión.

5.2.4 Módulo de control de encendido y apagado automático de luces.

En paralelo con el diseño anterior se diseñó este módulo el cual tiene como fin servir de iluminación artificial en momentos que la luz natural se acaba, para manipular el fotoperiodo de las plantas en investigaciones de GENFORES.

El módulo se encuentra compuesto por una tarjeta de desarrollo del fabricante Mikroelectronika, llamada MMB (Multimedia Board), en la cual a su vez, se encuentra un microcontrolador PIC32MX460F512L. En esta tarjeta se conectará la etapa de potencia, la cual alimenta a los halógenos usados como elemento de iluminación artificial. Esta tarjeta cuenta con una pantalla táctil para interacción con el usuario, cuenta con una memoria flash con capacidad de 512Kb para almacenar rutinas de software y datos, el microcontrolador tiene 85 pines dedicados a puertos E/S, los cuales son utilizados para acoplar los dispositivos periféricos que requiere el sistema.

Para controlar el encendido y apagado de las luces se utilizaron relés, los cuales al conectarse en serie con cada uno de las lámparas, abren o cierran el circuito.

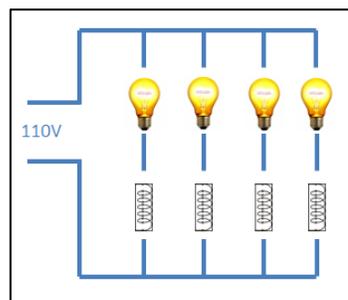


Figura 11. Control del encendido y apagado de luces por medio de relés

En la figura 11 se observa el diagrama general del módulo, que según los requerimientos el sistema, debía de contar con cuatro relés. Con base en el diagrama anterior, el siguiente paso en el diseño fue implementar la lógica de control de los relés. Para entender con más detalle, como se controlaron estos dispositivos, se muestra la figura 12.

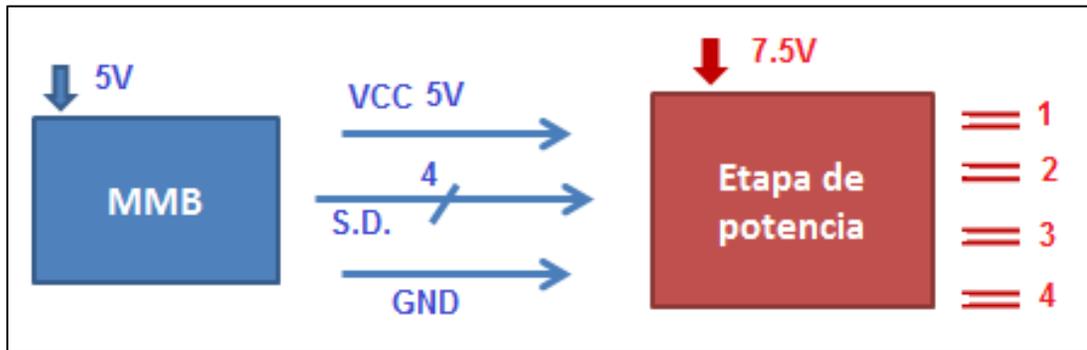


Figura 12. Conexión de la MMB con la etapa de potencia

La lógica general de control de este módulo se implementó a través de rutinas de software en la MMB, la cual con cuatro señales digitales en su salida actúa sobre los cuatro diferentes relés. Para acoplar estas señales digitales con el manejo de los relés se empleó una etapa de potencia, la cual en su entrada tiene una señal con la referencia de la MMB, otra con V_{CC} igual a 5V, y las 4 señales digitales. Además esta etapa requiere alimentarla con 7.5V externos. En su salida, simplemente se tiene las dos conexiones de los extremos de cada uno de los cuatro relés. Ahora, de qué trata esta etapa de potencia se observa en el siguiente diagrama esquemático, en la figura 13.

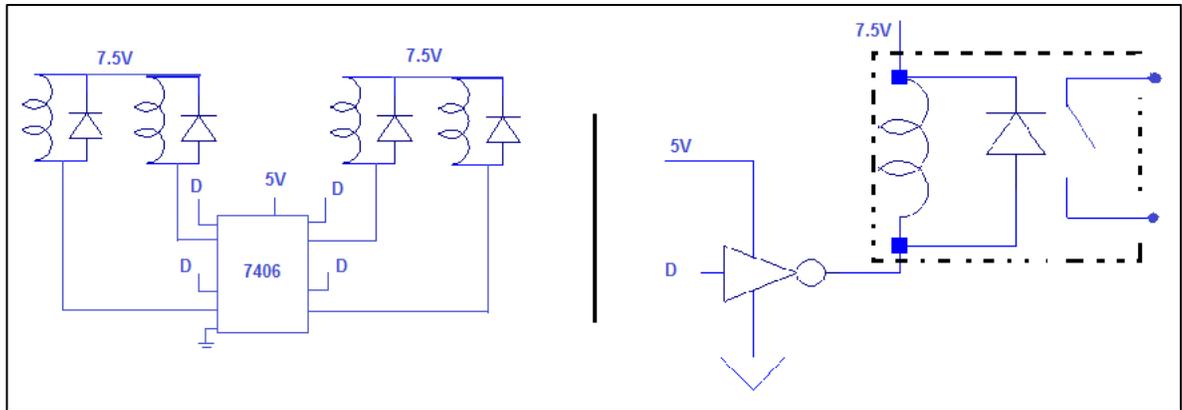


Figura 13. Etapa de potencia para control de los relés

Al lado derecho de la figura 13, se observa la etapa de potencia por la cual transcurre cada señal. Desde la MMB se envía una señal digital indicada con la letra “D” en el diagrama, con un estado lógico de 0 o 1 para actuar sobre el relé. Esta señal enviada pasa por el buffer de corriente 74LS06, el cual le da fuerza a la señal para alimentar la inductancia del relé, el cual está conectado a los 7.5V externos. Al lado izquierdo de la figura 13, se muestra la conexión del integrado 74LS06 con las 4 señales digitales, y los cuatro relés. Cabe mencionar que para alimentar este integrado se utiliza la señal de 5V proveniente de la MMB.

De esta manera se implementó el hardware del módulo en cuestión, aunado al diseño mostrado, como parte del montaje del sistema, se utilizó cable eléctrico para exteriores TGP 2x14 para las conexiones de las lámparas. También como parte de la implementación, la lámpara escogida fue el halógeno “Halopar 30 E26”, de la marca Osram, las cuales consumen 75W tienen una vida útil de 2000 horas.

5.2 Descripción del software

5.2.1 Rutinas de software implementadas sobre SIWA-RTOS

Las rutinas de software de los nodos se realizaron utilizando el ambiente de programación MPLAB IDE v8.80, la cual permite organizar de manera adecuada el ensamblador ASM32, el compilador C32 y el enlazador LINK32.

Las rutinas ejecutadas en los nodos se implementan sobre SIWA-RTOS, lo que implica programar las funciones que se deseen en forma de tareas. Para este proyecto no se modificó la estructura general del sistema operativo, simplemente se trabajó con las tareas, las interrupciones y las inicializaciones necesarias en cada caso.

La secuencia general que sigue la ejecución del programa para todos los nodos se observa en la figura 14 y corresponde a la programación del archivo main.c por lo que es la rutina principal.



Figura 14. Main del software de los nodos

En esta rutina como para todo microcontrolador, primero se le definen los bits de configuración, luego se realizan las inicializaciones necesarias, en este caso se crean las tareas, y posteriormente inicia el calendarizador el cual se encarga del manejo de las tareas.

5.2.2 Rutinas de software de los nodos RFD

En cuanto al software de los nodos RFD, tampoco se hicieron cambios significativos a la estructura de la programación ya que su funcionamiento ha sido bueno. Por otro lado realizaron cambios menores como remover las rutinas encargadas de leer datos de sensores y hacer cálculos de promedios, ya que no se encuentran conectados. Estas rutinas deben implementarse dependiendo del tipo de sensor que se le añada al sistema, por lo que varían según cada caso.

De modo que las rutinas seguidas en estos nodos siguen la secuencia que se muestra en el siguiente diagrama de flujo en la figura 15. La rutina mostrada se encuentra programada en el archivo taskMIWI.c.

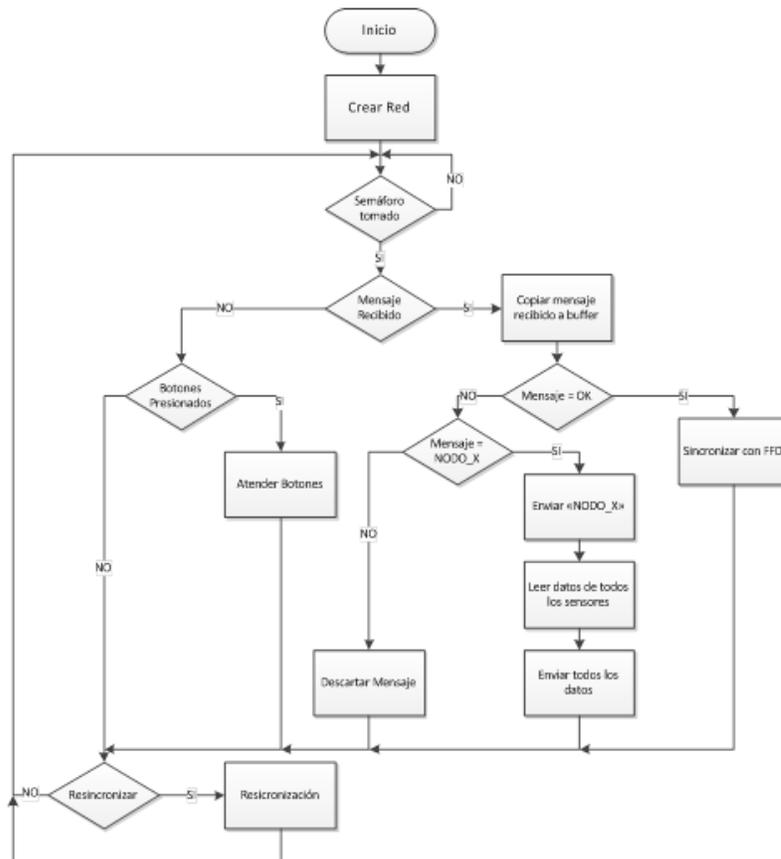


Figura 15. Diagrama de flujo de tarea taskMIWI en nodos RFD. [1]

La modificación que si se realizó fue en el momento en que los nodos constantemente se sincronizan con el FFD, ya que ellos envían el comando "CHECK", lo cual lo hacen periódicamente aproximadamente cada 10 segundos. Se añadió un comando adicional, ahora los nodos envían el CHECK y luego su identificación, para que el FFD reconozca cual nodo se encuentra en sincronía, por ejemplo si estamos hablando del nodo 2, este envía "CHECK" y luego "0x02". Esta rutina no se encuentra en el diagrama de flujo anterior ya que no depende de las condiciones ahí mostradas.

5.2.3 Rutinas de software del nodo FFD o nodo base

Diferente al caso anterior, las rutinas del nodo FFD cambiaron radicalmente. Para empezar, el sistema operativo realizaba en primera instancia tres tareas: MIWITask, STORAGETask y TCPIPTask. Como parte de las modificaciones se eliminaron las dos últimas tareas, ya que no era necesaria ni la comunicación vía Ethernet ni la comunicación vía USB para guardar información en una tarjeta de memoria. En la figura 16 se puede apreciar el flujo que sigue el programa en la tarea MIWITask, que se conservó pero se modificó.

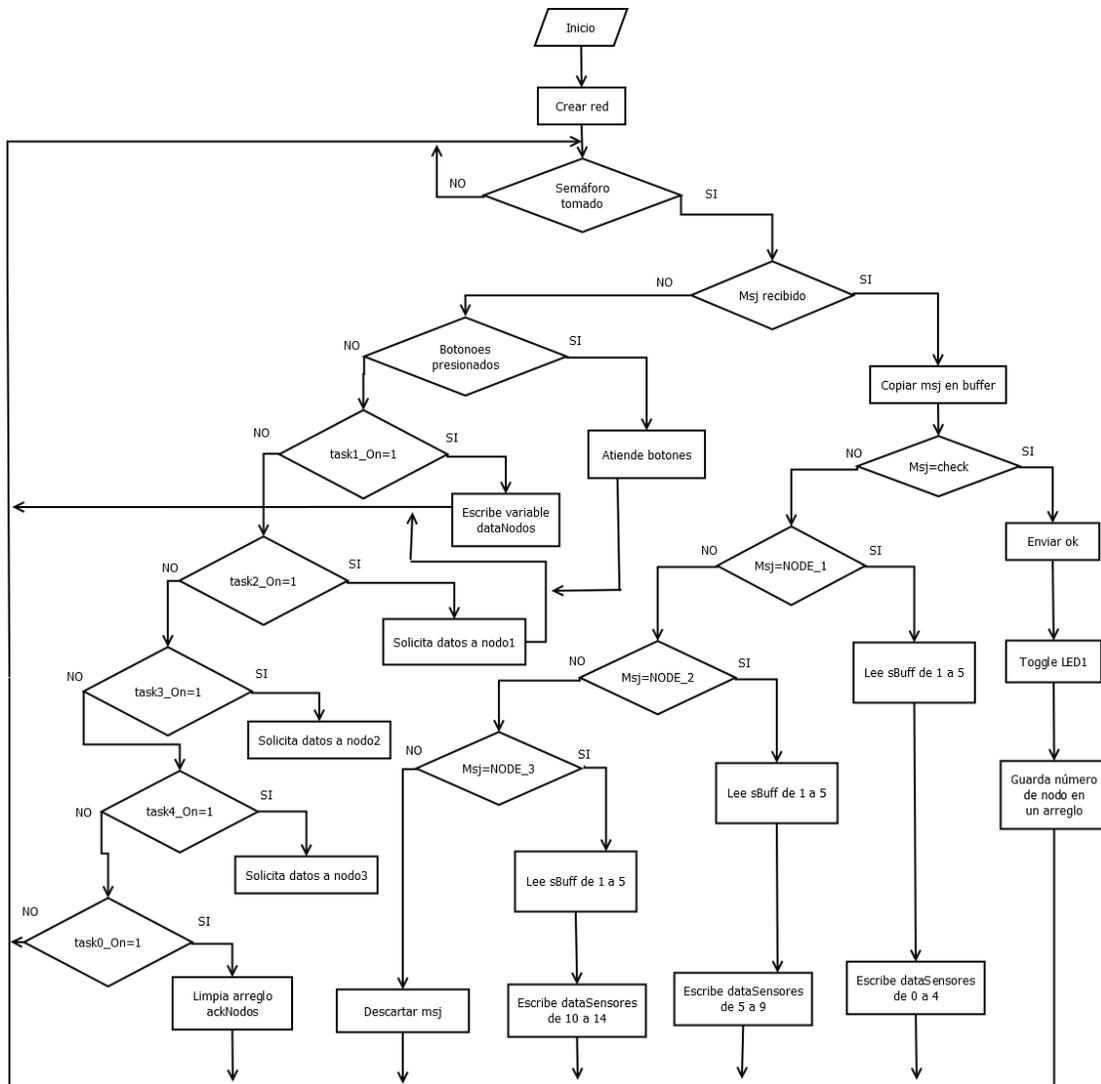


Figura 16. Diagrama de flujo de la tarea taskMIWI en nodo FFD

Hay varios aspectos a destacar en este diagrama, el primero es siempre se pregunta primero por si hay algún mensaje proveniente de los nodos, si es así se lee el buffer en entrada de la comunicación por Miwi, sino no es así se atienden una serie de condiciones controladas por variables. Si el mensaje de entrada proviene de un nodo primero se revisa si corresponde a un CHECK, si es así se responde con un “OK” y guarda en un arreglo el código del nodo que envió el mensaje para llevar un control de los nodos que se están reportando, después de realizar esto se varía el estado del LED1 como indicador visual de que los nodos se encuentran en sincronía con la red. Cuando el mensaje recibido no es para

chequeo, entonces es porque los nodos están enviando información de sus sensores porque así fue pedido por este mismo nodo FFD, por lo que la información se guarda en un arreglo llamado dataSensores.

Para este proyecto los botones no realizan ninguna acción, por lo que si se presionan no sucede nada. En cuanto a las variables task_On0, task_On1, task_On2, task_On3 y task_On4 estas son cambiadas a “1” por la interrupción del módulo I2C, dependiendo de los comandos que envíe la Beaglebone.

Tabla 4. Comandos empleados en la comunicación entre la Beaglebone y el PICMX795F512L

Comando BB	Variable que modifica	Descripción
0xAA	task1_On	Obtener variable dataNodos (nodos conectados a la red)
0xA1	task2_On	Solicitud de datos a Nodo 1
0xA2	task3_On	Solicitud de datos a Nodo 2
0xA3	task4_On	Solicitud de datos a Nodo 3
0xBB	dIndex	Limpia dIndex. (Índice del arreglo que escribe datos a la BB)
0xCC	task0_On	Limpia la variable con info de nodos conectados a la red

Las rutinas mostradas en la sección 5.2.3 se encuentran implementadas en el archivo taskMIWI.c del nodo FFD, sin embargo la modificación de las variables de la tabla 4, se realiza en la rutina de la interrupción, la cual se describe más adelante en la sección 5.2.7

5.2.4 Ubuntu y librerías necesarias para la implementación de la solución

Para trabajar con la Beaglebone, se instaló el sistema operativo Linux, más específicamente su distribución Ubuntu Oneiric 11.10. Para ello se instaló su imagen en una tarjeta SD, desde la cual se corre el sistema operativo en la BB.⁴

Esta tarjeta se programa utilizando un cable USB, el cual permite visualizar por medio de la consola la información desplegada por la BB. Para la instalación de Ubuntu y para acceder la tarjeta se siguieron una serie de pasos, desde la instalación de la imagen, hasta poder correr el sistema. [13]

Una vez el software instalado, se accede a la tarjeta por medio del comando:

```
screen /dev/ttyUSB1 115200
```

El cual permite acceder a la BB si la conexión se realizó por el cable USB, esto si en el directorio /dev de la PC se logra observar que se reconoció ttyUSB0 o ttyUSB1. Si no es así, se debe ejecutar el siguiente comando:

```
modprobe ftdi_sio vendor=0x0403 product=a6d0
```

Existe otra alternativa para acceder la BB desde la PC, y es a través de la conexión vía Ethernet y por medio de un enrutador, usando el comando:

```
ssh ubuntu@"dir..IP"
```

Donde dice dir...IP se debe colocar la dirección IP respectiva.

Una vez que se encuentra Ubuntu corriendo sobre la BB, es necesario actualizar una serie de paquetes lo cual se realiza conectando el cable Ethernet, y con conexión a internet se puede realizar la actualización. Para la implementación de la solución es necesario manejar el módem USB con la BB, por lo que Ubuntu cuenta con unas bibliotecas para este propósito. Se descargaron o actualizaron

⁴ A partir de aquí se utilizarán las siglas “BB” para referirse a la tarjeta BeagleBone.

las bibliotecas: ppp, minicom, sshpass, y man. La biblioteca ppp se refiere al protocolo de comunicación punto a punto, minicom es hyperterminal en Ubuntu, sshpass es un comando que permite realizar autenticaciones que solicitan ciertas páginas web, como por ejemplo la del servidor de electrónica, y por último man es una herramienta muy importante en el desarrollo de programas ya que es un manual de comandos.

Con las bibliotecas ya listas, el siguiente paso es la creación de los scripts, los cuales se ubicaron en el directorio /home de la BB, y se observan en la figura 17.

```
root@omap:/home# ls
a.sh  descargas  log.txt  SFTP.sh  updateTime.sh  uploadFile.sh
build localtime  pici2c.sh  ubuntu  upload
root@omap:/home#
```

Figura 17. Directorio /home de la BB con los scripts implementados

De colores verdes y encerrados en un cuadro rojo se identifican los archivos creados para este proyecto. A esta lista hay que añadirle 2 scripts más encargados del diálogo con el módem USB y uno encargado de la actualización de la fecha y hora. La carpeta upload subrayada con rojo, es donde se encuentran los archivos .csv generados para subir al servidor. Las funciones que realizan estos scripts se explican en las siguientes secciones.

5.2.5 Scripts utilizados en la Beaglebone para conectarse a internet por medio del módem USB y para conectarse con el servidor de electrónica

Para establecer in diálogo con el módem USB y establecer conexión a internet, fue necesaria la implementación de dos scripts. Uno de ellos se llamó chatGPRS1 ubicado en el directorio /etc/ppp, y el otro llamado GPRS ubicado en el directorio /etc/ppp/peers. Estos dos archivos son fundamentales, ya que en ellos se establecen todas las pautas y parámetros de esa comunicación. Ver apéndice 2.

Para establecer la comunicación por medio del protocolo ppp, usando GPRS es necesario primero contar con un archivo de configuración, y luego con uno que sirva como chat para hablar con el dispositivo. Estos dos archivos implementados cumplen con este requerimiento, con el script GPRS se configura el protocolo y con chatGPRS1 se le envían al módem los mensajes necesarios para conectarse a la red del ice. Un aspecto importante a mencionar, es que en algunos dispositivos es necesario por medio de minicom, utilizar el comando `AT^u2diag=0` para deshabilitar la opción CD (NAND-flashmemory drive) del módem, ya que puede causar problemas al intentar conectarse.

Otro detalle importante es que cuando se conecta el módem USB a la BB, en el directorio `/dev` debe corroborarse que se reconoció el dispositivo.

```
root@omap:/dev# ls tty*
tty    tty15  tty22  tty3    tty37  tty44  tty51  tty59  tty9    ttyS1
tty0   tty16  tty23  tty30  tty38  tty45  tty52  tty6    tty00   ttyS2
tty1   tty17  tty24  tty31  tty39  tty46  tty53  tty60  tty01   ttyS3
tty10  tty18  tty25  tty32  tty4    tty47  tty54  tty61  tty02   ttyUSB0
tty11  tty19  tty26  tty33  tty40  tty48  tty55  tty62  tty03   ttyUSB1
tty12  tty2    tty27  tty34  tty41  tty49  tty56  tty63  tty04   ttyUSB2
tty13  tty20  tty28  tty35  tty42  tty5    tty57  tty7    tty05
tty14  tty21  tty29  tty36  tty43  tty50  tty58  tty8    ttyS0
root@omap:/dev#
```

Figura 18. Módem USB conectado y reconocido por el sistema operativo

En el directorio `/dev` debe apreciarse lo encerrado en el cuadro rojo, como en la figura 18, sino los tres, por lo menos uno de ellos, que significa que se reconoció el módem.

5.2.6 Generación de archivos .csv

Los archivos que se suben al servidor que contienen toda la información recolectada por la red de sensores, son de extensión `.csv`, los cuales despliegan la información de manera más ordenada. En la BB se generan tres diferentes archivos con la información obtenida, uno es `fecha-data.csv`, el otro es `fecha-log.csv` y el último es un archivo `log.txt`. Este último se crea simplemente por

motivos de observar la trazabilidad del proceso, aunque no es parte de los requerimientos del sistema, como si lo son los otros dos archivos.

Una vez al día se suben estos archivos los cuales en su nombre tiene la fecha correspondiente. Los dos scripts responsables de subir los archivos al servidor son uploadFile.sh y SFTP.sh

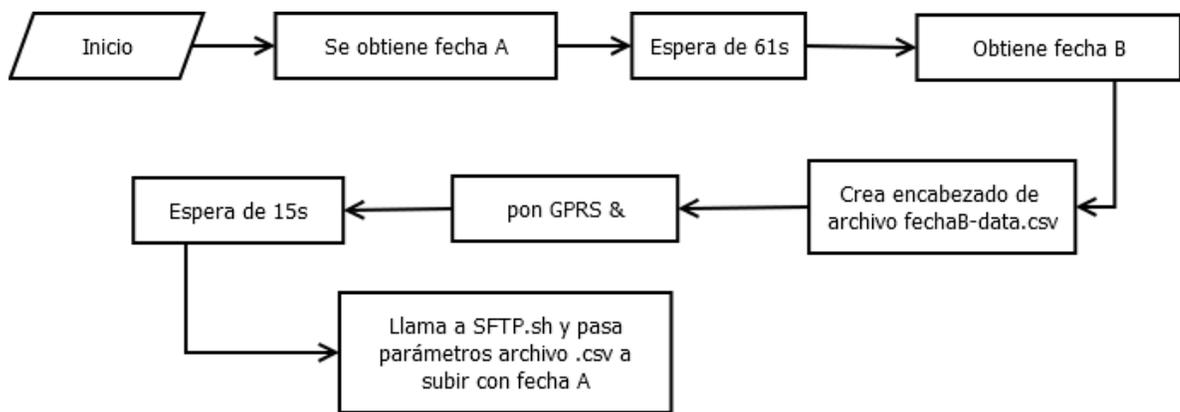


Figura 19. Diagrama de flujo de script uploadFile.sh

Como se observa en la figura 19, la rutina consta de obtener la fecha para darle nombre a los archivos que se suben cada día, este proceso se ejecuta a las 23:59, por lo que se espera 61s para que cuando se lee de nuevo la fecha, esta corresponde a la del siguiente día. De esta forma se suben los archivos con la fecha del día anterior, y se crea un archivo nuevo con la nueva fecha.

Para establecer la comunicación con el módem usando los scripts GPRS y chatGPRS1, se utiliza el comando:

pon GPRS &

De igual forma, para desconectarse de internet se usa el comando:

poff

Es así como en la rutina de uploadFile.sh después de crear el nuevo archivo, este se conecta a la red, llama al script SFTP.sh y le pasa como parámetros los archivos a subir. Su diagrama de flujo se observa en la figura 20.

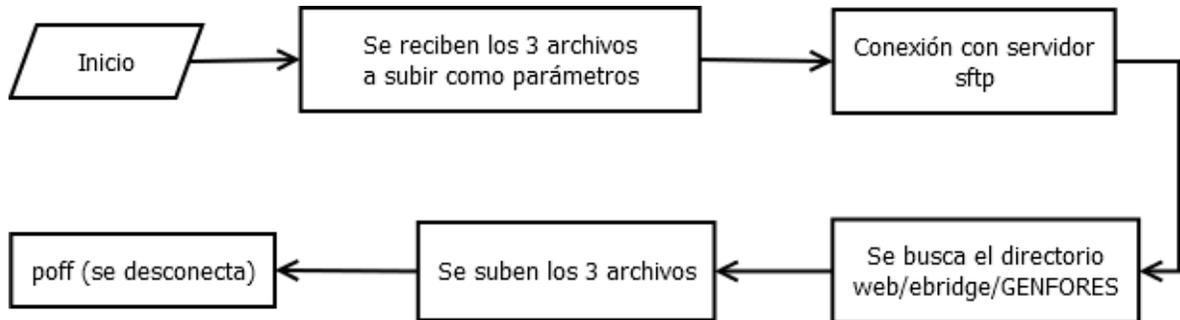


Figura 20. Diagrama de flujo de script SFTP.sh

El script SFTP.sh recibe como parámetros los tres archivos a subir, como ya se encuentra conectado a internet, intenta conectarse al servidor de electrónica por medio del comando:

```
sshpass -p 'password' sftp jcarvajal@www.ie.itcr.ac.cr
```

Posteriormente se sitúa en el directorio deseado y sube los archivos. Los archivos se suben usando el comando:

```
put
```

Finalmente se desconecta el módem se desconecta del servidor y de internet. Los pasos que se van siguiendo en cada una de las rutinas se van registrando en el archivo log.txt.

El script encargado de escribir propiamente la información respectiva en los archivos .csv es pici2c.sh, el cual está encargado de la comunicación por i2c y en paralelo se encarga de la escritura mencionada. En esta rutina se define la secuencia de la comunicación con el nodo base y la estructura que llevan los archivos a subir. En la figura 21 se aprecia más claramente el procedimiento seguido.

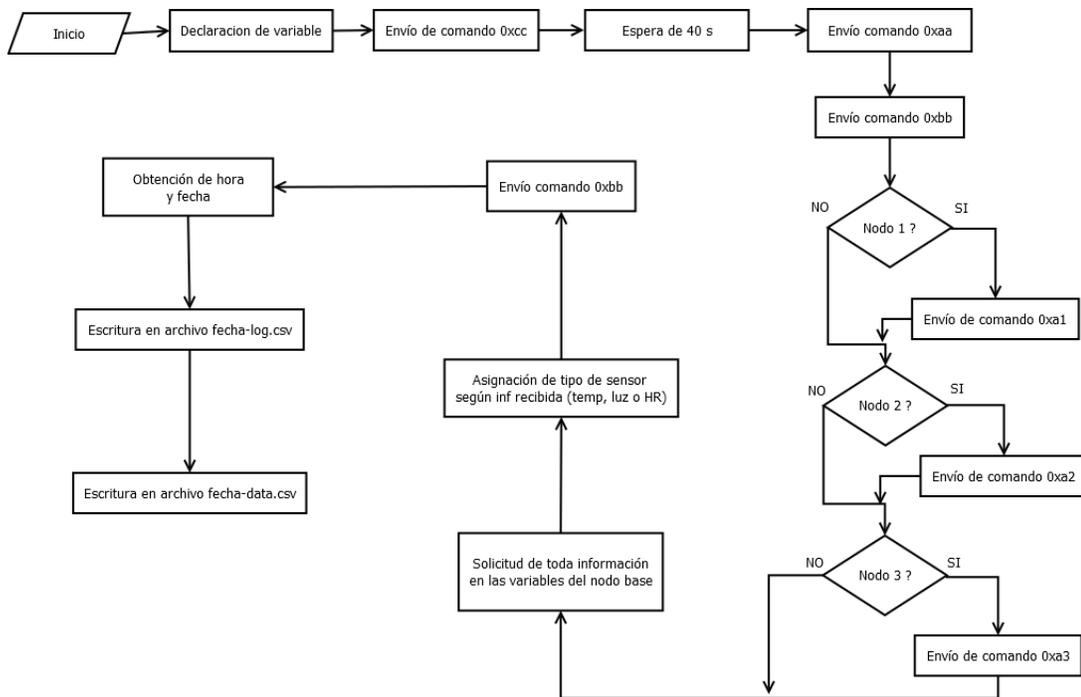


Figura 21. Diagrama de flujo de script pic2c.sh

Como se observa en la figura 21, primero se declaran las variables y los arreglos en donde se almacenará la información recibida por i2c, posteriormente se envía el comando 0xcc para solicitar al nodo base que inicie revisión de sincronía con sus nodos. Se espera 40 s para darle tiempo al nodo sumidero de recibir comprobaciones de sincronía de los nodos. Con el comando 0xaa se solicita el dato de nodos conectados, y dependiendo de esta información se hace la solicitud de datos solo de los nodos que se encuentran conectados. El comando 0xbb limpia una variable que sirve de índice para el envío de datos por i2c la cual se encuentra en la programación del PICMX795F512L. Con la información ya recolectada por el nodo base, se solicitan todos los datos con los valores de los sensores y tipos de sensores. Luego de esto, se limpia de nuevo el índice, se obtienen la fecha y hora del sistema, y en este punto de la rutina con toda la información almacenada en las variables del script pic2c.sh, se escribe sobre los archivos .csv.

5.2.7 Software implementado en la Beaglebone y en el PIC32MX795F512L para establecer la comunicación por medio del protocolo I2C

Antes de establecer la comunicación, se definió a la BB como el dispositivo maestro, y al nodo base como el dispositivo esclavo del protocolo I2C, de modo que solo la BB inicia la comunicación por esta vía, y por lo tanto dirige la comunicación por medio de comandos anteriormente explicados en la tabla 4.

Del lado de la BB, primero se verificó cuales módulos I2C se encontraban disponibles para su uso, y en el directorio /dev se observa que estaban habilitados para usar I2C1 e I2C3, como en la figura 22.

```
root@omap:/dev# ls i2c*  
i2c-1 i2c-3  
root@omap:/dev#
```

Figura 22. Módulo I2C habilitados en la Beaglebone

Según se hicieron las conexiones en el hardware, se utilizó el módulo I2C3. Para realizar la comunicación por medio de este protocolo bastaba con 3 diferentes comandos:

- a) *i2cdetect -y 3*
- b) *i2cset -y 3 0x20 0x0 0xaa*
- c) *i2cget -y 3 0x20 0x0*

El primero sirve para detectar en que direcciones se encuentra conectado un dispositivo esclavo, donde se obtuvo que en la dirección 0x20 se encontraba conectado el nodo base, así definido en la inicialización de las rutinas del nodo.

El segundo comando se emplea para escribirle al dispositivo esclavo con dirección 0x20, usando el módulo 3 el dato 0xaa en este caso. El parámetro “-y” se emplea para ejecutar la escritura sin que el sistema operativo solicite confirmación del usuario. Y por último el tercer comando se emplea para solicitar datos al esclavo, en este caso también al localizado en la dirección 0x20.

En la BB resulta simple emplear el protocolo, sin embargo en el PIC el procedimiento es un poco más complejo. Del lado del nodo base o dispositivo esclavo el protocolo se trabajó a nivel de interrupciones, por lo que solo se respondía a peticiones del maestro.

Para inicializar el módulo I2C en el PIC, se programó el registro I2C2CON=0x8340, que implica modo maestro sin activar, tamaño de la dirección del dispositivo de 7 bits, velocidad de respuesta desactivada, activación de STREN para esperar respuesta si es necesario, llamado general desactivado y activación de obediencia ante el SMBus. Se le asignó la dirección 0x20 al registro I2C2ADD, y muy importante se enmascaró la dirección escribiendo 0x20 en I2C2MSK. Por último se habilitaron las interrupciones y se le limpiaron las banderas correspondientes.

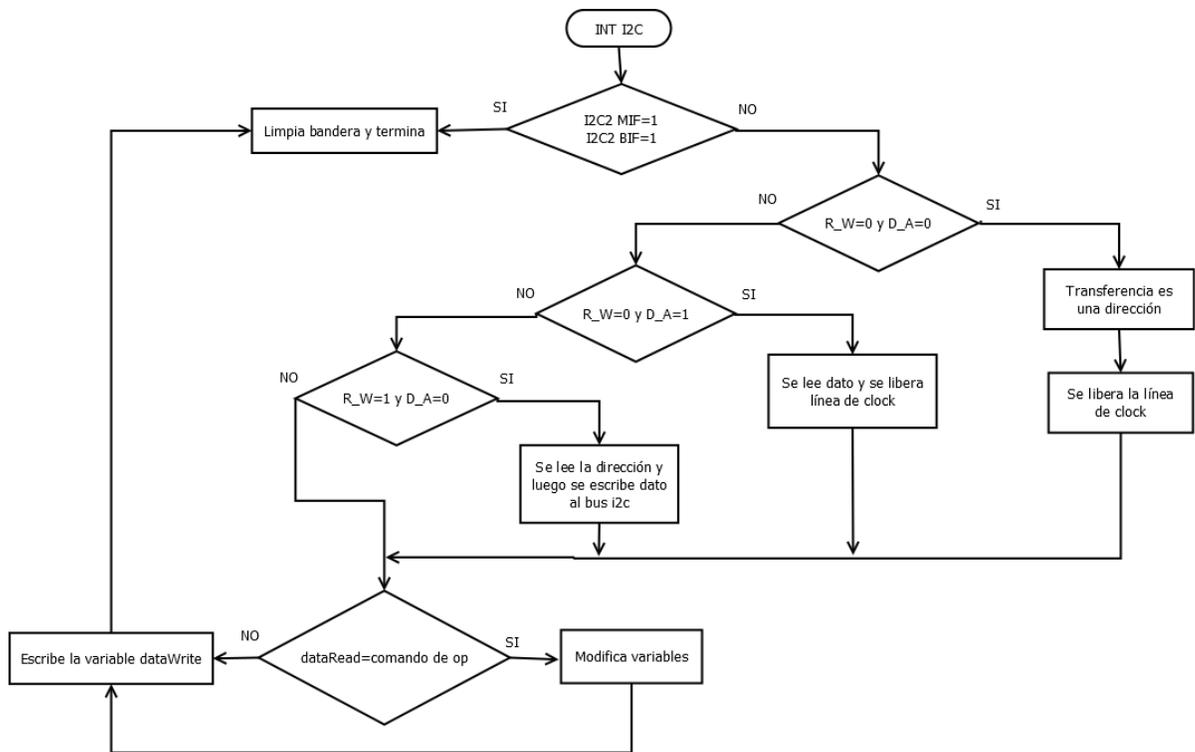


Figura 23. Rutina para comunicación por I2C implementada en el PIC32 del nodo sumidero

La secuencia seguida por la rutina que atiende la interrupción se observa en la figura 23, destacando que la interrupción se genera cuando el esclavo identifica su dirección en el bus de comunicación. Primero se revisan banderas para descartar cualquier colisión en el bus, si no hay ningún problema es necesario determinar si el dato en el bus corresponde a una dirección, si es información y si se va a realizar una escritura o lectura del maestro. Si se solicita una escritura del maestro, se modifican las variables según la tabla 4. Si se requiere de una lectura de la BB, se envía la información del dato del arreglo de salida apuntado por la variable dIndex.

5.2.8 Actualización de fecha y hora

Para la actualización de la fecha y la hora se utiliza el script updateTime.sh, el cual se ejecuta siempre que se inicia Ubuntu, y luego se ejecuta periódicamente según se desee. Para este proyecto se realiza el ajuste una vez por semana. Su diagrama de flujo se detalla en la figura 24.

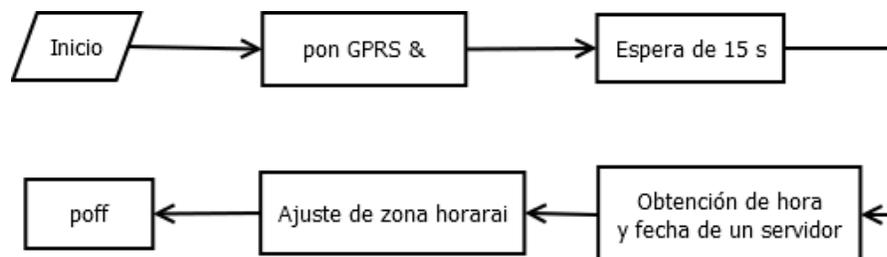


Figura 24. Diagrama de flujo de script updateTime.sh

Los pasos seguidos corresponden a la conexión del módem a internet, se espera 15 s para darle tiempo al dispositivo de establecer es conexión, se obtiene la hora y fecha de un servidor en la red usando el comando:

```
ntpdate cr.pool.ntp.org
```

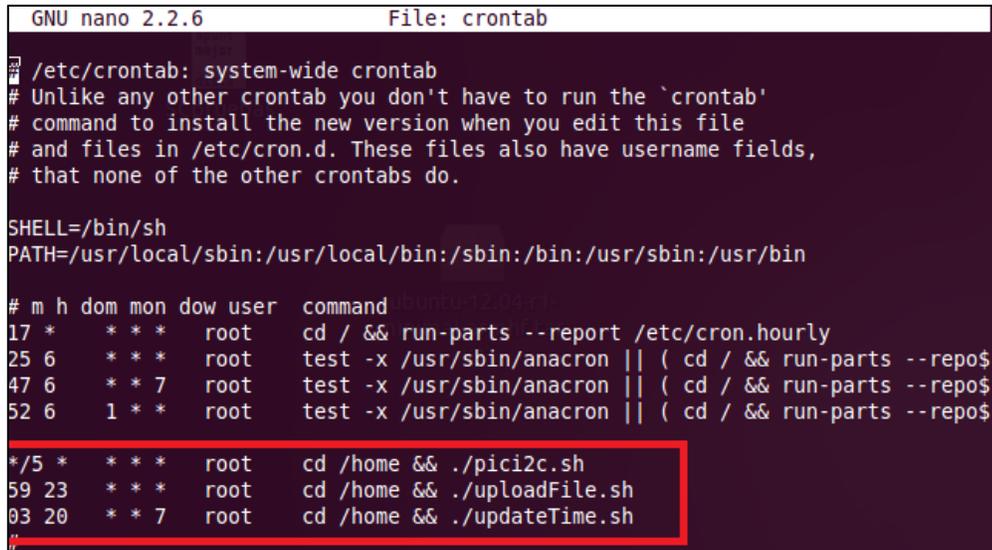
Posteriormente se ajusta la zona horaria con el comando:

```
ln -sf /usr/share/zoneinfo/America/Costa_Rica /etc/localtime
```

Finalmente, se desconecta de internet y se termina con la ejecución.

5.2.9 Ejecución programada de las tareas.

Para que los scripts creados se ejecuten de forma automática y periódica, se escribe en el archivo “crontab” el cual se encuentra en el directorio /etc



```
GNU nano 2.2.6 File: crontab
/etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$
47 6 * * 7 root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$
52 6 1 * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --repo$

*/5 * * * * root    cd /home && ./pici2c.sh
59 23 * * * root    cd /home && ./uploadFile.sh
03 20 * * 7 root    cd /home && ./updateTime.sh
#
```

Figura 25. Archivo crontab

En el cuadro rojo de la figura 25 se encierra la programación de los tres scripts a ejecutar. El primero es pici2c.sh encargado de la petición de los datos al nodo base, se ejecuta cada minutos. El script uploadFile.sh que sube la información a la red se ejecuta todos los días a las 23:59, y a las 00:01 este mismo llama al SFTP.sh. Por último se escogió una hora aleatoria para actualizar la fecha y hora, y el script updateTime.sh se ejecuta los domingos a las 20:03.

5.2.10 Rutinas de software implementadas en la MMB

Las rutinas de software en la MMB esencialmente consisten en las siguientes etapas:

- Inicialización de la tarjeta de desarrollo
- La interfaz gráfica con el usuario
- Control de encendido y apagado

En esta etapa se configura los componentes de la tarjeta de desarrollo así como los pines del microcontrolador PIC32. Se creó un archivo llamado HardwareProfile.h el cual asigna un nombre a los pines que conectan los componentes que contiene la tarjeta de desarrollo con el microcontrolador a excepción de la pantalla táctil, la memoria Flash y memoria EEPROM los cuales se asignan en archivos aparte, aunque se encuentre implementado en la misma tarjeta de desarrollo (TouchScreen.c (.h), SerialFlash.c (.h) y SerialFlash.c (.h), respectivamente). Por otro lado, en el archivo principal Main.c, se configura los diferentes registros dentro del microcontrolador.

En cuanto a la interfaz gráfica con el usuario, se utilizó la herramienta para programación en C para el microcontrolador PIC32 MPLAB IDE, propiedad de Microchip, en esta herramienta se instaló el plug-in Graphics Display Designer (GDD) [14], el cual es útil para el diseño de interfaces gráficas. Dicha herramienta utiliza las librerías gráficas disponibles de la empresa Microchip. Al terminar el diseño, se le indica a la herramienta que genere el código en C. Se genera tres archivos .c con sus respectivos archivos en extensión .h: GDD_GraphicsConfig.c, GDD_Resources.c (.h) y GDD_Screens.c. (.h).

En la figura 26 se observa algunas pantallas diseñadas mediante la herramienta GDD para el módulo de control de encendido y apagado automático de luces.



Figura 26. Ventanas diseñadas mediante la herramienta Graphics Display Designer

Con respecto al sistema principal se implementó un sistema al que se le ajusta la fecha y la hora cada vez que se enciende. El sistema puede controlar hasta cuatro diversos túneles, entendiendo túnel por salida con diferentes configuraciones (4 halógenos con distintas configuraciones). También se le pueden programar 5 diferentes perfiles, los cuales corresponden a 5 diferentes intervalos de tiempo los cuales pueden ser pre-configurados. Entonces para asignarle un tiempo de encendido y apagado a una de las salidas, se debe hacer un emparejamiento entre un perfil y un túnel, de este modo ese túnel se encenderá y se apagará a la horas que tiene fijadas ese perfil de tiempo.

Entonces, el usuario puede realizar diversas funciones, puede reajustar la fecha y la hora, puede cambiar las contraseñas ya que se requiere una para hacerle modificaciones al sistema, puede ajustar los perfiles, puede guardar el historial si inserta una tarjeta SD, o asignarle a un túnel un perfil para que este se encienda y apague según se desea.

Capítulo 6: Análisis de resultados

En este capítulo se muestran los resultados obtenidos a lo largo de todo el proyecto, y se analiza por sección cada una de las acciones emprendidas en la búsqueda de la solución.

6.1 Diagnostico del sistema a mejorar realizado al inicio del proyecto

El diagnóstico realizado al inicio del proyecto necesario para comprender las aristas del problema requirió desplazarse al invernadero ubicado en Santa Clara, dicha evaluación arrojó los siguientes resultados.

Tabla 5. Resultados de evaluación de diagnóstico inicial del sistema

Aspecto evaluado	Estado	Comentarios
Nodo sumidero	Funcionando	Presenta calentamiento al tacto
Memory Stick (Nodo sumidero)	Dañada	No es posible accederla
Nodo1	Dañado	Se encontraba con agua en las conexiones
Nodo2	Funcionando	-
Nodo3	Funcionando	-
Enrutador inalámbrico	Dañado	No enciende
Sensores	15 elementos dañados	No responden a los nodos, ni estímulos externos
Conexiones (cable UTP)	En buen estado	Las conexiones con cable para exteriores en muy buen estado
Conectores de alimentación	2 en buen estado y 3 dañados	Conector del enrutador quemado, conector del nodo 1 quemado, y conector de alimentación tipo USB con corrosión.

Analizando estos resultados y contrastándolos con el problema de que no se tenía acceso a la información desde la página web, varios aspectos influían. No se iba a subir la información porque el enrutador inalámbrico estaba quemado. Esto

sugiere prestar atención a irregularidades en la calidad de la corriente eléctrica del invernadero. Por lo que se deben tomar previsiones como el uso de reguladores de voltaje. Además conectando el nodo sumidero vía Ethernet directamente con otro enrutador y con una PC, tampoco hubo respuesta de la página web, que si se levantó, pero no cargó datos. En primer lugar no se tenía acceso a la información de los sensores porque estos no estaban funcionando, los nodos 2 y 3 si estaban transmitiendo información al nodo base, y la comunicación vía Ethernet si estaba funcionando puesto que se levantaba la página web, sin embargo esta causaba un calentamiento significativo en el nodo base. El nodo 1 se dañó debido a una mala instalación, por falta de mantenimiento o falta de cuidado, ya que se desprendió de donde se encontraba, se hallaba en el suelo, y por ende le entró agua a sus circuitos. Por lo que no solo debe procurarse el buen funcionamiento electrónico de un sistema, sino también su instalación o implementación. Otro punto es que las conexiones con el cable UTP con el que se conectaban los sensores a los nodos se encontraban en buen estado, y bien hechas, por lo que no es un punto débil del sistema. No así el encapsulado de los sensores, los cuales se encontraban dañados, tirados en el suelo, cubiertos de tierra y agua.

Otro factor que influía para que la información de los parámetros no llegara a visualizarse, es el estado de la memoria USB usada, la cual estaba dañada ya que no se podía escribirle ni leerle información, ni siquiera desde una PC.

6.2 Evaluación de las condiciones operativas actuales y el uso de tecnología más robusta para aplicaciones de monitorización de ambientes extremos para mejora de la vida útil de los sensores

Este diagnóstico mostró que las condiciones más críticas en cuanto a parámetros ambientales difíciles para los sistemas electrónicos, las experimentan los sensores, ya que se encuentran dentro de los jardines clonales con altas temperaturas y una muy alta humedad relativa, ya que ahí es donde se experimenta con las plantas efectuando un riego y aislándolas del ambiente con

un plástico el cual las cubre. Los nodos RFD se encuentran cerca de estos puntos de riego, pero se encuentran a condiciones ambientales normales. En cuanto al nodo sumidero, este si se encuentra protegido de la lluvia o el riego que se de, y más lejos de los jardines donde la temperatura es alta, por lo que los cuidados en el encapsulado debe priorizarse con los sensores.

Evaluando las condiciones del lugar, analizando las diferentes opciones que ofrece el mercado en términos en cuanto a calidad de sensores, se encontró que existe gran cantidad de opciones en cuanto a tecnologías de medición, sensores que ofrecen menores rangos de incertidumbre, mayor confiabilidad, diversas formas de obtención de la información (digitales o analógicos), diferentes acondicionamientos de señal, entre otros factores, sin embargo en cuanto a lo robusto que resulta su implementación en ambientes con condiciones difíciles, el factor determinante no es en gran medida la tecnología del chip, sino el tipo de encapsulado o las protecciones que se le dan a ese sensor. Por esta razón, se recomienda que para la nueva implementación de sensores en este invernadero se utilicen sensores con encapsulados herméticos, los cuales no permiten que les afecte el agua o la humedad. Sin dejar de lado, de encapsular y sellar cualquier circuito acondicionador de señal que se implemente con ellos, además de buscar ubicar los sensores en los puntos donde se encuentren bien fijados, y no sufran caídas o golpes. Se recomienda el uso como sensor de humedad relativa un sensor como el HM1500-ND de Digkey [15], el cual no le afecta la inmersión en agua por ejemplo y se encuentra en un encapsulado apto para las condiciones que puede experimentar. Para sensor de temperatura se recomienda el uso de un sensor como el LM35CAH-ND [16] pero muy importante con encapsulado TO46-3 el cual es hermético y lo protege de condiciones adversas como abundante humedad, y altas temperatura ya que soporta hasta 150°C. Como sensor de luz también se recomienda el uso de fotodiodos o fototransistores que de igual forma que el sensor de temperatura, cuenten con encapsulados herméticos, como con el fotodiodo BPX65, TO18 [17]. Cabe mencionar que una medición con un fotodiodo requiere acondicionamiento de señal, por lo que debe procurarse proteger bien el

circuito, y utilizar un fotodiodo que trabaje para el rango de lux (iluminancia) que se desea cuantificar.

6.3 Red de sensores: Nodos FFD y RFD (Protocolo Miwi)

La evaluación dio como resultado que de los tres nodos RFD implementados solo uno dejó de funcionar, y esto se debió a que se desprendió de donde se encontraba, y al encontrarse en el suelo lo alcanzó agua de lluvia o del riego propio del lugar. Pero en cambio los dos nodos RFD se encontraban funcionando plenamente a pesar de las condiciones del lugar de la ausencia de mantenimiento. Además el nodo FFD también se encontraba en buen funcionamiento, ya que creaba la red Miwi al que se asocian los otros nodos, y además ejecutaba la tarea de comunicación por Ethernet, aunque ésta resultara en un sobrecalentamiento del dispositivo. Este aspecto, brinda confiabilidad en el hardware empleado, hablando propiamente de las tarjetas Starter Kit con sus respectivas tarjetas de expansión, y de la funcionalidad de los microcontroladores PICMX360F512L de los nodos RFD y del PICMX795F512L del nodo FFD, que si requirió depurar su software. De igual forma es confiable la utilización de los transceptores MRF24J40, junto con la estabilidad del sistema operativo SIWA-RTOS para levantar la red Miwi.

Para discutir los efectos de los cambios realizados en la implementación del sistema, se debe primero observar los resultados de mediciones de consumo de corriente. Cambios que implicaron remover la comunicación Ethernet, la tarjeta de memoria USB y el enrutador, para cambiar el modelo de la comunicación añadiendo un dispositivo como la Beaglebone para usar un módem USB y así utilizar la infraestructura de telecomunicaciones, para acceder a la información desde cualquier punto con internet.

6.3.1 Mediciones de corriente en Nodo RFD

En este apartado se describen las mediciones realizadas para obtener una caracterización del efecto de diversas funciones en el consumo de corriente de la red.

Para efectuar las mediciones de corriente se efectuaron tres ensayos diferentes. En todos los casos se midió la corriente versus el tiempo con periodo de muestreo de 18ms, que permitió el instrumento.

En el primer ensayo se conectó el nodo FFD a 12V fijos, sin utilizar la comunicación I2C y conectándole uno de los nodos. El proceso realizado fue: se alimentó el nodo, aproximadamente a los 7s el nodo terminó de crear la red ya que él apaga los LED's 0 y 2 cuando termina con ese proceso, alrededor de los 25s se alimentó el nodo RFD 3 por aparte al otro lado del laboratorio, luego aproximadamente a los 45s se tiene certeza que hay comunicación entre los nodos ya que en el proceso de chequeo el FFD varía el estado del LED 1. Al minuto se termina la medición.

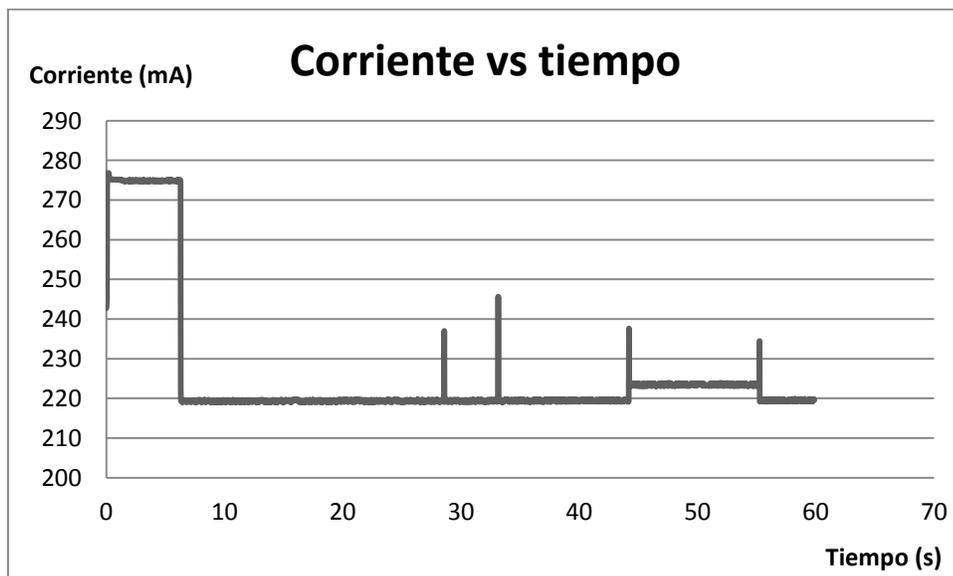


Figura 27. Primer ensayo para medición de corriente en nodo FFD

Observando la figura 27 se aprecia el incremento en el consumo de corriente mientras se levanta la red (primeros 7s), también cuando se conectan los nodos

RFD a la red se dan dos picos de corriente, y cada vez que los nodos RFD envían el CHECK para verificación de sincronía, en el FFD se da un pico de corriente como se aprecia aproximadamente a los 45s y 55s. Otro punto importante a destacar es que mientras el LED 1 se encuentra encendido, se aprecia la diferencia en el consumo de corriente, que sucede en la medición entre los 45s y los 55s.

En el segundo ensayo se conectó el nodo FFD a los mismos 12V fijos. El proceso realizado fue: se alimentó el nodo, aproximadamente a los 7s el nodo terminó de crear la red, alrededor de los 17 s se alimentó el nodo RFD 3 por aparte al otro lado del laboratorio, luego aproximadamente a los 45s se tiene certeza que hay comunicación entre ambos nodos ya que FFD varía el estado del LED 1. Aproximadamente a los 55s la BB envía un comando por I2C (0xcc), espera 40s y a los 95s se envían los demás comandos y se recibe la información del nodo base. A los 2 minutos se termina la medición.

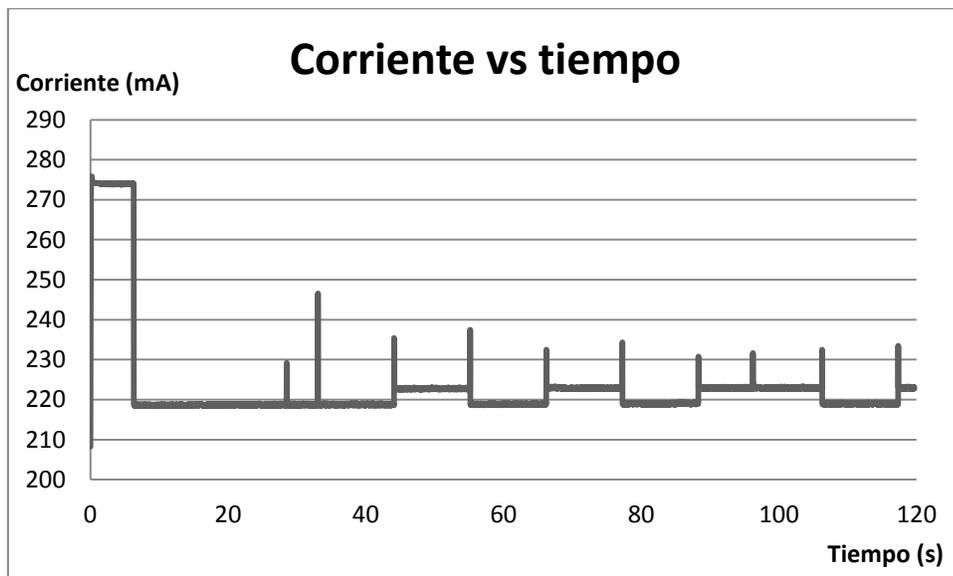


Figura 28. Segundo ensayo para medición de corriente en nodo FFD

Observando los resultados en la figura 28, al igual que en el primer caso, se observa el elevado consumo de corriente a la hora de la creación de la red, los 2 picos de corriente entre los 20s y los 40s se dan mientras el RFD se sincroniza con la red. Y los siguientes picos de corriente se dan por intervalos aproximados

de 10s, correspondientes al chequeo del Nodo RFD, donde se realiza la conmutación del LED 1. Mientras el LED 1 permanece encendido se aprecia la diferencia en el nivel bajo de corriente. Un detalle importante, es que se aprecia un pico de corriente a aproximadamente a los 95s, el cual coincide con el momento en el que se envía toda la información del nodo base a la BB por I2C.

En el tercer ensayo se contrasta la medición de corriente del nodo FFD conectándole dos nodos a la red: RFD 2 y RFD 3, contra la medición de corriente del nodo FFD con las rutinas antes de emplear la optimización, en otras palabras, con la programación que tenía al inicio de este proyecto.

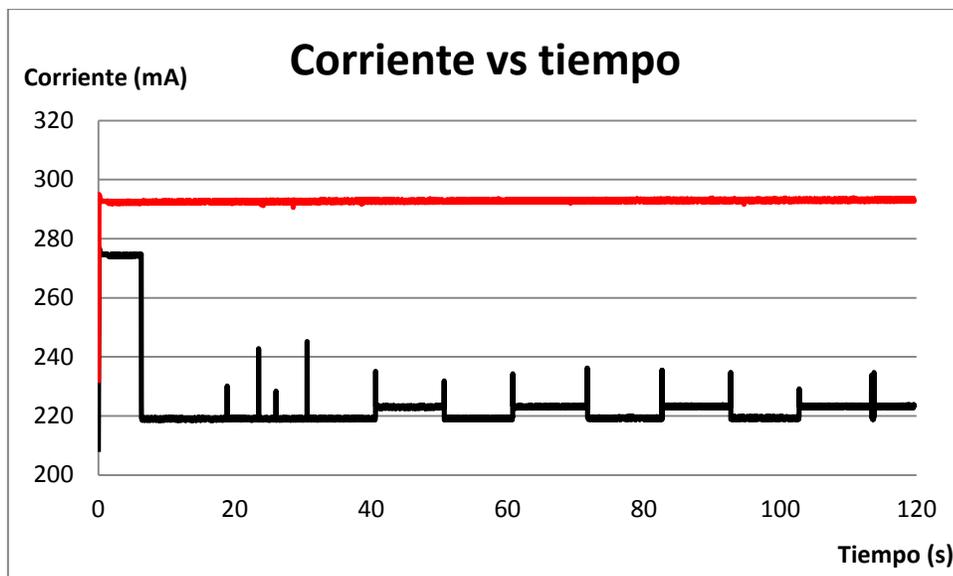


Figura 29. Tercer ensayo para medición de corriente en nodo FFD

En la figura 29 se observa en color negro el nodo con la optimización implementada, y en rojo antes de implementarla. Se observa como ahora la corriente media en su operación es cercana o aproximada a los 220mA, mientras que anteriormente llegaba casi a los 300mA. Cabe mencionar que debido a que se efectuaron modificaciones en rutinas de software en los nodos RFD también, la medición de corriente en rojo no sufre perturbaciones ya que en esa medición no se conectaron a la red los nodos RFD, ni se emplea un LED como elemento visual para determinar si los nodos están en sincronía o no. Además aunque sus tareas de USB y Ethernet están activas a nivel de software, realmente no están usándose

esos módulos completamente ya que se removieron la tarjeta USB de memoria y no hay un enrutador para que el FFD se comunique con él, lo que implicaría incrementos en el consumo de corriente.

Los resultados mostrados muestran la efectiva reducción en el consumo de corriente en el nodo base, y por ende la disminución de su carga de trabajo. Además sugieren que la comunicación por I2C no implica cambios significativos en el consumo de corriente del sistema. Sin embargo, el LED implementado como indicador visual del reporte de sincronía de los nodos RFD, si evidencia el incremento en el consumo.

Equipo utilizado: Para las mediciones se utilizó el equipo Unidad Fuente Medidor de la marca Keithley 2636A. La configuración del equipo para la medición fue la siguiente. Con el botón “dis” se seleccionó visualizar el canal A, con el valor de la fuente en voltios y su límite de corriente entregada. Con el botón “src” y manipulando la perilla, se fija la tensión de la fuente. Con el botón “lim” se fija el límite de corriente que entrega la fuente. Con el botón “meas” se escoge medir corriente. Luego para el almacenamiento de datos se seleccionó el botón “config”, haciendo uso de la perilla se escoge la opción “common”, luego “store”. Allí se selecciona como “storage mode” la opción “overwrite”, y se indica que los datos se guardan en el “chanAbuff”. En la opción “dest” que aparece cuando se selecciona chanAbuff, se escoge “chanAbuff1”. También ahí mismo se selecciona “buffer1” y es necesario activar la opción “clear” para limpiar ese buffer. Después en el mismo lugar en la opción “elements” se pone “TSTAMP” en “ON” para obtener en los datos el tiempo de las mediciones. Por último fuera ya de la configuración para obtener la curva, se presiona “store” y luego “take readings”, allí se escoge el número de muestras y se le da “enter”, para iniciar las mediciones se enciende el canal conectado “CHA ON/OFF”.

6.4 Uso de la Beaglebone y modem USB

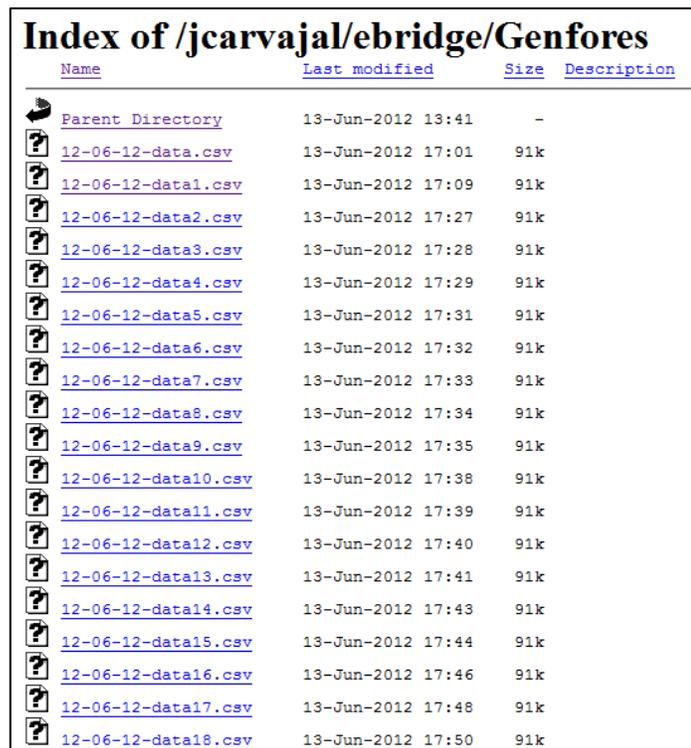
La implementación de la Beaglebone se dio con el fin de hacer uso del módem USB a través de las librerías de Linux, más específicamente de Ubuntu Oneiric 11.10. Esta tarjeta no solo permite correr sobre Linux, sino que su tamaño compacto y bajo costo representó una buena opción para implementar la solución como se hizo. Por otro lado, se escogió usar el módem USB Huawei E173, debido a la facilidad de adquirirlo por su distribución en el país por parte del ICE-Kolbi. Además de ello, analizando los diferentes planes ofrecidos por las empresas de telecomunicaciones, resultó para los fines de este proyecto más adecuado emplear un chip (SIM) Kolbi debido a que el pago de internet se cobra por minuto, y no por día, semana, o mes como en Movistar o Claro. Además la intensidad de la señal en el invernadero con chip Kolbi es buena.

6.5 Resultados de los scripts de subida y para comunicación I2C usados en la Beaglebone

Para comprobar el funcionamiento general del sistema diseñado, se puso el sistema a trabajar continuamente por más de un día para que recolectara toda la información durante 24 horas cada 5 minutos. Al final de esto, se generó el archivo 12-06-12-data.csv y el archivo 12-06-12-log.csv. Tienen ese nombre porque la prueba se realizó en el Instituto Tecnológico de Costa Rica con sede en Santa Clara, el día 12 de junio del 2012. De esa manera se probó subir los archivos con la cantidad máxima de información la cual es la recolectada durante 24 horas, porque después de ese tiempo se crea otro archivo diferente. El peso máximo alcanzado por el archivo –data.csv fue de 91k.

Después de que la Beaglebone recolectó datos del nodo base durante más de un día, con el archivo creado, se probó subir al siguiente día continuamente a la red desde el invernadero el archivo ya generado y almacenado en la Beaglebone con el fin de establecer una estadística de intentos fallidos y aciertos.

En la figura 30 se muestra la prueba realizada.



<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory	13-Jun-2012 13:41	-	
 12-06-12-data.csv	13-Jun-2012 17:01	91k	
 12-06-12-data1.csv	13-Jun-2012 17:09	91k	
 12-06-12-data2.csv	13-Jun-2012 17:27	91k	
 12-06-12-data3.csv	13-Jun-2012 17:28	91k	
 12-06-12-data4.csv	13-Jun-2012 17:29	91k	
 12-06-12-data5.csv	13-Jun-2012 17:31	91k	
 12-06-12-data6.csv	13-Jun-2012 17:32	91k	
 12-06-12-data7.csv	13-Jun-2012 17:33	91k	
 12-06-12-data8.csv	13-Jun-2012 17:34	91k	
 12-06-12-data9.csv	13-Jun-2012 17:35	91k	
 12-06-12-data10.csv	13-Jun-2012 17:38	91k	
 12-06-12-data11.csv	13-Jun-2012 17:39	91k	
 12-06-12-data12.csv	13-Jun-2012 17:40	91k	
 12-06-12-data13.csv	13-Jun-2012 17:41	91k	
 12-06-12-data14.csv	13-Jun-2012 17:43	91k	
 12-06-12-data15.csv	13-Jun-2012 17:44	91k	
 12-06-12-data16.csv	13-Jun-2012 17:46	91k	
 12-06-12-data17.csv	13-Jun-2012 17:48	91k	
 12-06-12-data18.csv	13-Jun-2012 17:50	91k	

Figura 30. Prueba de subida de archivos al servidor de electrónica

Desde la Beaglebone se intentó subir intentándolo solo una vez 19 archivos diferentes continuamente, todos con el mismo peso (91k), y el resultado fue que los 19 archivos se subieron al servidor sin problemas, tal y como se observa en la figura 30.

La comunicación por I2C resultó ser una buena solución para comunicar la BB y el microcontrolador ya que estos se encuentran a una distancia muy corta (no más de 5cm), lo cual es un factor a favor en las líneas de este protocolo. Además el protocolo cuenta con señales de ACK, las cuales permiten confirmar si la información enviada llegó de correctamente. También su implementación no representa un costo en consumo de corriente para el nodo base, y su implementación es muy cómoda en la Beaglebone.

Según las especificaciones requeridas, el archivo fecha-data.csv a subirse una vez por día debía mostrar información de la hora a la que se tomó a medición, el nombre del sensor, el tipo del sensor, el valor leído, las unidades de esa cantidad física y el nodo del cual proviene. En la figura 31 se muestra es resultado del archivo creado abriéndolo con la herramienta Excel.

	A	B	C	D	E	F
1	Hora	NodoID	Sensor	Tipo	Valor	Unidad
2	00:05:01	N2	s21	HumedadRelativa	0x21	%
3	00:05:01	N2	s22	Luz	0x22	lux
4	00:05:01	N2	s23	Temperatura	0x23	°C
5	00:05:01	N3	s31	Temperatura	0x31	°C
6	00:05:01	N3	s32	Temperatura	0x32	°C
7	00:05:01	N3	s33	HumedadRelativa	0x33	%
8	00:05:01	N3	s34	Luz	0x34	lux
9	00:05:01	N3	s35	Luz	0x35	lux
10	00:10:01	N2	s21	HumedadRelativa	0x21	%
11	00:10:01	N2	s22	Luz	0x22	lux
12	00:10:01	N2	s23	Temperatura	0x23	°C
13	00:10:01	N3	s31	Temperatura	0x31	°C
14	00:10:01	N3	s32	Temperatura	0x32	°C
15	00:10:01	N3	s33	HumedadRelativa	0x33	%
16	00:10:01	N3	s34	Luz	0x34	lux
17	00:10:01	N3	s35	Luz	0x35	lux
18	00:15:01	N2	s21	HumedadRelativa	0x21	%
19	00:15:01	N2	s22	Luz	0x22	lux
20	00:15:01	N2	s23	Temperatura	0x23	°C

Figura 31. Archivo fecha-data.csv subido al servidor

De igual forma se solicitó crear un archivo fecha-log.csv, con la información de los nodos conectados a la red y sus respectivos sensores, como se ve en la figura 32.

	A	B	C
1	Nodo	Nombre de sensor	Descripcion
2	2	s21	HumedadRelativa
3	2	s22	Luz
4	2	s23	Temperatura
5	3	s31	Temperatura
6	3	s32	Temperatura
7	3	s33	HumedadRelativa
8	3	s34	Luz
9	3	s35	Luz

Figura 32. Archivo fecha-log.csv subido al servidor

En estos 2 archivos se resume toda la información recolectada por la red de sensores, transmitida a la Beaglebone y después subida a internet.

6.6 Implementación del módulo de control de encendido y apagado automático de luces

El hardware usado en esta implementación se escogió debido que se encontraba a disposición en primer lugar, porque significa una solución que brinda una interfaz amigable para el usuario, que le resulta más fácil manejar un módulo el cual resulta muy intuitivo. Además, la MMB permite emplear un control más completo del sistema permitiendo la implementación de varias funciones, y ofreciendo la opción de no tener que agregar dispositivos adicionales para su manipulación como teclados o displays. La etapa de potencia usada busca proteger el circuito electrónico de la tarjeta MMB de las diferencias en los niveles de tensión con respecto a los niveles de los halógenos controlados. Los halógenos escogidos fueron buscados con el fin de brindar la cantidad de flujo luminoso (lum) suficiente para engañar al fotoperiodo de las plantas, y de ser aptos para ambientes húmedos en exteriores. El efecto de este control sobre la fotosíntesis de las plantas se pretende caracterizar con mediciones de luz con los sensores por implementar en este sistema de caracterización de parámetros climáticos optimizado.

Se debe mencionar que aunque el diseño se realizó para controlar la luz en cuatro túneles, a la hora del montaje en el lugar solo se conectaron tres debido a requerimientos del personal de GENFORES. Sin embargo, aunque el diseño pueda realizar el control de cuatro diferentes configuraciones, se pueden añadir más elementos en paralelo a los halógenos para aumentar la cantidad de lámparas, teniendo en cuenta que la limitante es la potencia que puedan disipar los relés usados, los cuales pueden ser cambiados ya que su montaje así lo permite.

En las figuras 33 y 34 se muestra el sistema montado y conectado en el invernadero de GENFORES, con sus cajas respectivas para protección y para interacción con el usuario en el caso de la pantalla táctil.



Figura 33. Sistema de control de encendido y apagado automático montado y funcionando en el invernadero



Figura 34. Iluminación montada y funcionando en el invernadero

Capítulo 7: Conclusiones y recomendaciones

7.1 Conclusiones

1. El consumo de corriente en el nodo base (FFD) no es afectado significativamente cuando se da la comunicación por medio del protocolo I2C con la Beaglebone.
2. Las tareas removidas del sistema operativo SIWA-RTOS correspondientes a la comunicación por Ethernet y por USB con una tarjeta de memoria, disminuyeron fuertemente el consumo de corriente en el nodo FFD y por ende su sobrecalentamiento.
3. La nueva arquitectura implementada permite tener acceso a los datos de la red de sensores vía internet desde cualquier punto accediendo al servidor de la Escuela de Ingeniería Electrónica del ITCR.
4. El empleo de sensores en ambientes exteriores debe considerar como un factor de suma relevancia el tipo de encapsulado con el que cuenta y las protecciones físicas que se le añadan.
5. La arquitectura de la red CRTECMote es confiable debido al funcionamiento óptimo de sus nodos ante condiciones climáticas difíciles cuando la instalación es la apropiada.
6. El módulo con interfaz táctil para el usuario agregado permite implementar estrategias de control de luminosidad adecuadas para la operación del mini jardín-clonal

7.2 Recomendaciones

1. En el empleo de sistemas electrónicos sensibles a cambios bruscos en la tensión de alimentación como lo es este, es adecuado la utilización de reguladores de voltaje.
2. Ante la instalación de equipo electrónico en ambientes con condiciones adversas para el mismo, un mantenimiento del sistema aunque sea mínimo puede prolongar la vida útil de sus componentes en gran medida.
3. Para el uso de sensores en ambientes húmedos, debe procurarse buscar encapsulados herméticos que no permitan la entrada de agua a los circuitos.
4. El consumo de corriente de los nodos puede reducirse si así se desea implementando rutinas periódicas para poner en bajo consumo al microcontrolador o a los mismos transceptores.
5. El sistema implementado opera completamente de forma automática, pero si así se desea se puede emplear mensajería de texto para darle instrucciones al sistema, aprovechando el módem USB.
6. Se pueden Implementar rutinas adicionales para reinicio del sistema y para la limpieza de la memoria en la tarjeta Beaglebone, así también de detección de errores como MD5 checksum.

Bibliografía y referencias

- [1] Fallas, J. P. (2011). *Caracterización de parámetros climáticos en mini jardines clonales forestales bajo ambiente protegido. (Proyecto de graduación de licenciatura)*. Cartago, Costa Rica: Instituto Tecnológico de Costa Rica.
- [2] Carvajal, J., Araya, F., Navarro, A., Pérez, D., Moreira, C., Estrada, M., & Herrera, J. (2011). Diseño de un nodo con arquitectura abierta para aplicaciones con redes inalámbricas de sensores (CRTECMote).24(N 2.), P. 27-33. Obtenido de http://www.tec.ac.cr/sitios/Vicerrectoria/vie/editorial_tecnologica/Revista_Tecnologia_Marcha/pdf/tecnologia_marcha_24-2/3.pdf
- [3] Ian, F., A., Weilian, S., Yogesh, S., & Erdal, C. (2002, A survey on sensor networks. *Georgia: IEEE Communications Magazine*, Obtenido de <http://meslab.snu.ac.kr/courses/dip2010f/papers/10-Akyildiz02.pdf>
- [4] Álvarez, E. N. (s.f.). *Mediciones de temperatura, termómetros e introducción*. Obtenido 02/21, 2012, de <http://laboratorios.fi.uba.ar/lscm/termometros.pdf>;
- [5] Hernández, L. E., Pineda, W. & Bayona, D. A. (2088). *Sistema de control de humedad y temperatura para invernaderos*. Obtenido 02/20, 2012, de http://www.edutecne.utn.edu.ar/microcontrol_congr/comunicaciones/Sistema_control_humedad.pdf;
- [6] Perdigones, A., Peralta, I., Nolasco, J., Muñoz, M. & Pascual, V. (2004). *Sensores para el control climático en invernadero*. Obtenido 02/22, 2012, de http://www.magrama.gob.es/ministerio/pags/biblioteca/revistas/pdf_Hort/Hort_2004_179_44_51.pdf;
- [7] Papalias, T. A., & Wong, M. (2006). *Making sense of light sensors*. Obtenido 02/20, 2012, de <http://www.eetimes.com/design/automotive-design/4009874/Making-sense-of-light-sensors>;
- [8] Yifeng, Y. (2009). *Microchip wireless application interface - MiApp*. Obtenido 04/27, 2012, de <http://ww1.microchip.com/downloads/en/AppNotes/01284A.pdf>;
- [9] *Modem USB*. (s.f.). Obtenido 04/09, 2012, de <http://www.modemusb.com/>
- [10] Cruellas, J. (2000). *Sistemas de transporte de datos*. Obtenido 04/11, 2012, de http://people.ac.upc.edu/cruellas/std/Tema_1_2_2000_20001_q1.PDF;
- [11] *Sistema GPRS*. (s.f.). Obtenido 04/13, 2012, de www.uv.es/~montanan/redes/trabajos/GPRS.doc;
- [12] Quintal, M. C. (s.f.). *El bus I2C*. Obtenido 05/01, 2012, de http://www.google.co.cr/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&ved=0CGwQFjAC&url=http%3A%2F%2Fgalia.fc.uaslp.mx%2F~cantocar%2Fmicrocontroladores%2FEL_BUS_I2C%2FEL_BUS_I2C.PPT&ei=g3HbT5SdLYWS9gT8n8D9Cg&usq=AFQjCNH_7e9aCBvNpbmiotQHMQB0BxOUsg&sig2=sSk1f5o6;
- [13] *BeagleBoard Ubuntu*. (2012). Obtenido 04/27, 2012, de <http://elinux.org/BeagleBoardUbuntu>;

- [14] *Microchip graphics display designer*. (s.f.). Obtenido 04/26, 2011, de http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2680&dDocName=en544475;
- [15] *Relative humidity module*. (2001). Obtenido 03/10, 2012, de <http://media.digikey.com/pdf/Data%20Sheets/Humirel%20PDFs/HM1500%5B1%5D.pdf>;
- [16] *LM35 precision centigrade temperature sensors*. (2000). Obtenido 03/10, 2012, de <http://www.ti.com/lit/ds/symlink/lm35.pdf>;
- [17] *BPX 65*. (2010). Obtenido 03/10, 2012, de <http://catalog.osram-os.com/catalogue/catalogue.do?favOid=000000200017b0900180023&act=showBookmark>;
- [18] Coley, G. (2012). *Beaglebone rev A5 system reference manual*. Obtenido 04/23, 2012, de http://beagleboard.org/static/beaglebone/latest/Docs/Hardware/BONE_SRM.pdf;
- [19] Tarrío, P. (2008). *Redes de sensores fundamentos y aplicaciones*. Obtenido 06/05, 2012, de <http://www.slideshare.net/yerartSlide/uimp-redes-de-sensores-fundamentos-y-aplicaciones>;
- [20] Microchip. (s.f.). *Starter kit I/O expansion board*. Obtenido 06/15, 2012, de http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2615&dDocName=en535444;
- [21] Microchip. (s.f.). *Miwi development enviroment*. Obtenido 06/15, 2012, de http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2113¶m=en520414;
- [22] MikroElektronika. (2009). *MultiMedia board for PIC32MX4*. Obtenido 09/12, 2011, de http://ww1.microchip.com/downloads/en/DeviceDoc/pic32mx4_mmb_manual_v100.pdf;
- [23] Barry, R. (2009). *Using the FreeRTOS real time kernel: A practical guide*. Obtenido 02/16, 2011, de <http://www.lulu.com/shop/richard-barry/using-the-freertos-real-time-kernel-a-practical-guide-pic32-edition/paperback/product-14727276.html>;
- [24] Cisco Systems, I. (2009). *Point to point protocol*. Obtenido 04/24, 2012, de http://docwiki.cisco.com/wiki/Point-to-Point_Protocol;
- [25] *Configuring a dialer*. (2010). Obtenido 04/26, 2012, de <http://ernstagn.home.xs4all.nl/GPRS-HOWTO/GPRS-HOWTO-6.html>;
- [26] Hitachi Semiconductor. (1999). *74LS06 datasheet*. Obtenido 03/15, 2012, de http://www.datasheetcatalog.net/es/datasheets_pdf/7/4/0/6/7406.shtml;
- [27] *Laboratorio de sistemas electrónicos para la sostenibilidad (SESLab)*. (2011). Obtenido 10/23, 2011, de <http://www.seslab.ietec.org/index.html>;
- [28] Microchip Technology Inc. (2008). *Inter-integrated circuit 61116D*. Obtenido 05/07, 2012, de <http://ww1.microchip.com/downloads/en/DeviceDoc/61116D.pdf>;
- [29] Microchip Technology Inc. (2011). *PIC32MX5XX/6XX/7XX family data sheet*. Obtenido 02/25, 2011, de <http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en545660>;

- [30] Murillo, O., Obando, G., Badilla, Y. & Azofeifa, M. (2009). *Creación de GENFORES, una cooperativa de mejoramiento genético forestal en costa rica*. Obtenido 10/23, 2011, de http://www.una.ac.cr/inisefor/index.php?option=com_remository&Itemid=0&func=startdown&id=39;
- [31] *Planes kolbi datos - datacard*. (2012). Obtenido 04/12, 2012, de http://www.grupoice.com/wps/portal!/ut/p/c4/04_SB8K8xLLM9MSSzPy8xBz9CP0os3gig2CT4BBHX8cgVxNLA08fX39jy5AQQwsjQ_2CbEdFAGGzrMU!;
- [32] Vindas, L. (2009, dic. 04). Lunes inicia ventas de líneas 3G en costa rica. *El Financiero*, Obtenido de http://www.elfinancierocr.com/ef_archivo/2009/diciembre/06/tecnologia2183746.html

Apéndices

A.1 Glosario, abreviatura y simbología

RTOS: (*Real Time Operating System en inglés*), es un sistema operativo que ha sido desarrollado para aplicaciones de tiempo real.

SPI: (*Serial Peripheral Interface en inglés*), es un estándar de comunicaciones usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos de forma serial.

GPRS: (*General Packet Radio Services en inglés*). Servicio general de paquetes vía radio.

MCU: (*Microcontroller Unit en inglés*). Microcontrolador.

Consola: Linux utiliza una herramienta llamada consola de comandos o Terminal desde la cual se pueden ejecutar una serie de sentencias para acceder a ciertos lugares del sistema, realizar ciertas tareas o simplemente visualizar reportes importantes.

BB: Beaglebone.

TCP/IP: Protocolo de control de transmisión (TCP) y Protocolo de Internet (IP).

RFD: (*Reduced-function device en inglés*). Nodos de una red de área personal que no pueden ser coordinadores.

FFD: (*Full-function device en inglés*). Coordinador de una red de área personal.

MODEM (Modulador-Demodulador): es un periférico utilizado para transferir información entre varios equipos a través de una técnica de transmisión y recepción de señales eléctricas denominada Modulación y Demodulación.

A.2 Archivos GPRS y chatGPRS1

Archivo GPRS:

```
/dev/ttyUSB0
115200
noauth
connect "/usr/sbin/chat -v -f /etc/ppp/chatGPRS1"
defaultroute
usepeerdns
modem -detach
fin***
```

Archivo chatGPRS1:

```
ABORT 'NO CARRIER'
ABORT 'VOICE'
ABORT 'NO DIALTONE'
ABORT 'BUSY'
ABORT 'ERROR'
ABORT 'NO ANSWER'
ABORT 'DELAYED'

"" 'ATZ'

OK 'AT&F'
OK 'ATQ0 V1 E1'
OK 'AT&D2 &C1'
OK 'AT+FCLASS=0'
OK 'ATS0=0'

OK 'AT+CGDCONT=1,"IP","kolbi3g"'
OK 'ATDT*99***1#'
CONNECT '\c'
Fin***
```