



**UNIVERSIDAD CATÓLICA DE SANTIAGO DE
GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL
DESARROLLO CARRERA DE INGENIERÍA ELECTRÓNICA EN
CONTROL Y AUTOMATISMO

TEMA:

Aplicación del protocolo de comunicación KNX en el control de
luces de una vivienda unifamiliar

EL AUTOR:

Dávila Chevalier Alejandro Andrés

Trabajo de Titulación previo a la obtención del título de
INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO

TUTOR:

Ing. Córdova Rivadeneira, Luis Silvio MSc.

Guayaquil, Ecuador

20 de Marzo del 2019



**UNIVERSIDAD CATÓLICA DE SANTIAGO DE
GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL
DESARROLLO

CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

CERTIFICACIÓN

Certificamos que el presente trabajo fue realizado en su totalidad por el Sr. Dávila Chevalier, Alejandro Andrés como requerimiento para la obtención del título de INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO.

TUTOR

Ing. Córdova Rivadeneira, Luis Silvio MSc.

DIRECTOR DE CARRERA

Ing. Heras Sánchez, Miguel Armando

Guayaquil, a los 20 del mes de marzo del año 2019



UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL
DESARROLLO**

**CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO**

DECLARACIÓN DE RESPONSABILIDAD

Yo, Dávila Chevalier Alejandro Andrés

DECLARO QUE:

El trabajo de titulación “Aplicación del protocolo de comunicación KNX en el control de luces de una vivienda unifamiliar” previo a la obtención del Título de Ingeniero Electrónico en Control y Automatismo, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría. En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 20 del mes de marzo del año 2019

EL AUTOR

Dávila Chevalier, Alejandro Andrés



UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL
DESARROLLO

CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO

AUTORIZACIÓN

Yo, **Alejandro Andrés Dávila Chevalier**

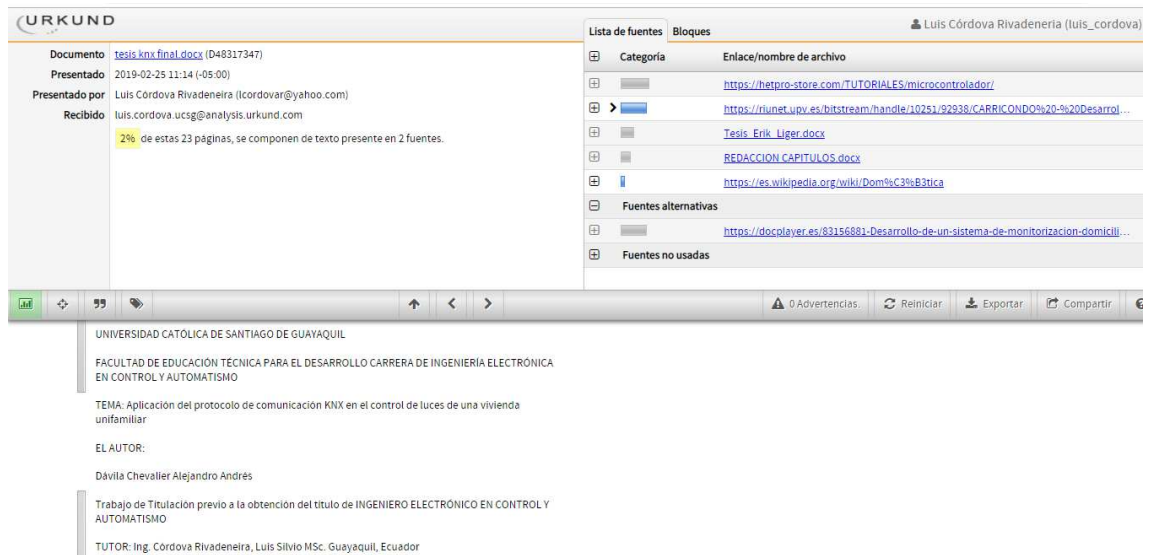
Autorizo a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: “Aplicación del protocolo de comunicación KNX en el control de luces de una vivienda unifamiliar”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

EL AUTOR

ALEJANDRO ANDRÉS DAVILA CHEVALIER

REPORTE DE URKUND

Informe del trabajo de Titulación de la Carrera de Ingeniería Electrónica en Control y Automatismo, con el 2% de coincidencia perteneciente al estudiante DAVILA CHEVALIER ALEJANDRO ANDRES.



The screenshot displays the URKUND interface. On the left, document details are shown: 'Documento' is 'tesis knx final.docx (D48317347)', 'Presentado' is '2019-02-25 11:14 (-05:00)', 'Presentado por' is 'Luis Cordova Rivadeneira (lcordova@yahoo.com)', and 'Recibido' is 'luis.cordova.ucsg@analysis.orkund.com'. A yellow highlight indicates '2% de estas 23 páginas, se componen de texto presente en 2 fuentes.' On the right, a 'Lista de fuentes' (List of sources) table is visible, listing various URLs under categories like 'Fuentes alternativas' and 'Fuentes no usadas'. The bottom of the interface shows a navigation bar with icons for search, back, forward, and other functions, along with a status bar indicating '0 Advertencias' and options for 'Reiniciar', 'Exportar', and 'Compartir'.

UNIVERSIDAD CATÓLICA DE SANTIAGO DE GUAYAQUIL
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y AUTOMATISMO
TEMA: Aplicación del protocolo de comunicación KNX en el control de luces de una vivienda unifamiliar
EL AUTOR:
Dávila Chevallier Alejandro Andrés
Trabajo de Titulación previo a la obtención del título de INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO
TUTOR: Ing. Cordova Rivadeneira, Luis Silvio MSc. Guayaquil, Ecuador

Atte.

Ing. Luis Córdova Rivadeneira MSc.

Tutor

DEDICATORIA

El presente trabajo está dedicado a mi familia que me han apoyado desde que comencé mis estudios universitarios, especialmente a mi abuelo, Sr. Fermín Ludgerio Chevalier Ronquillo que gracias a sus sabios consejos pude seguir adelante con mis estudios.

A mi madre, Arq. Marlene Alexandra Chevalier Jaramillo por su apoyo durante toda mi carrera universitaria.

EL AUTOR

ALEJANDRO ANDRÉS DAVILA CHEVALIER

AGRADECIMIENTO

Agradezco sobre todo a mi familia que me han apoyado en toda mi carrera universitaria y que han estado conmigo en todo momento importante que ha pasado en mi vida, también a la Universidad Católica Santiago de Guayaquil por permitirme adquirir conocimientos en tan prestigiosa entidad.

A mis docentes, que me guiaron durante los años de mi carrera, para llegar a ser un profesional en el área en la que me especializo, y también por los conocimientos que me han brindado mediante teoría y práctica.

A mis compañeros, aquellos que conocí en las aulas de clase y con quienes compartí momentos muy especiales.

A todas las personas que conocí y me apoyaron en mi formación académica y han hecho posible la ejecución de este trabajo.

EL AUTOR

ALEJANDRO ANDRÉS DÁVILA CHEVALIER



**UNIVERSIDAD CATÓLICA DE SANTIAGO DE
GUAYAQUIL**

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL
DESARROLLO**

**CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y
AUTOMATISMO**

TRIBUNAL DE SUSTENTACIÓN

f. _____

ING. ROMERO PAZ, MANUEL DE JESÚS

DECANO

f. _____

ING.MENOZA MERCHAN, EDUARDO VICENTE

COORDINADOR DE TITULACIÓN

f. _____

ING. NINO VEGA URETA M.Sc.

OPONENTE

Índice General

| | |
|---|-----------------|
| ÍNDICE DE FIGURAS..... | XII |
| ÍNDICE DE TABLAS | XIII |
| RESUMEN | XIV |
| ABSTRACT..... | XV |
| <u>CAPÍTULO 1: DESCRIPCIÓN DEL TRABAJO DE TITULACIÓN</u> | <u>2</u> |
| 1.1. INTRODUCCIÓN. | 2 |
| 1.2. JUSTIFICACIÓN..... | 3 |
| 1.3. PLANTEAMIENTO DEL PROBLEMA..... | 3 |
| 1.4. OBJETIVOS DE LA INVESTIGACIÓN | 4 |
| 1.4.1 OBJETIVO GENERAL..... | 4 |
| 1.4.2. OBJETIVOS ESPECÍFICOS. | 4 |
| 1.5. METODOLOGÍA DE LA INVESTIGACIÓN | 4 |
| 1.6. ESTADOS DEL ARTE..... | 5 |
| 1.6.1. SISTEMA DOMÓTICO PARA UNA CASA INTELIGENTE | 5 |
| 1.6.2. DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE VIVIENDA DOMÓTICA BASADO EN LAS PLATAFORMAS ARDUINO Y ANDROID | 5 |
| 1.6.3. DISEÑO DE UN MÓDULO PARA EL CONTROL DE ILUMINACIÓN DE UN HARDWARE DOMÓTICO A TRAVÉS DE UN TELÉFONO CELULAR EN UN AMBIENTE DISTRIBUIDO | 6 |
| 1.6.4. ESTUDIO DE PREFACTIBILIDAD PARA IMPLEMENTAR SISTEMAS DOMÓTICOS EN LAS CASA DEL SECTOR DE LA ESPAÑOLA EN LA CIUDAD DE BOGOTÁ PARA POTENCIAR EL BIENESTAR HUMANO Y EL MEDIO AMBIENTE. | 6 |
| 1.6.5. DISEÑO E IMPLEMENTACIÓN DE UNA CASA DOMÓTICA PARA PERROS DOMÉSTICOS USANDO TECNOLOGÍA RASPBERRY PI Y ARDUINO CON CONTROL A TRAVÉS DE INTERFAZ WEB. | 7 |
| 1.6.6. GESTIÓN DOMÓTICA DE UNA CASA UNIFAMILIAR BASADA EN ARDUINO..... | 8 |
| 1.6.7. DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE DOMÓTICA BASADO EN LA TECNOLOGÍA SMART BUS KNX PARA EL CONTROL DE ILUMINACIÓN, AUDIO Y SEGURIDAD, MEDIANTE UN ENLACE WEB APPS | 8 |
| 1.6.8. DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA BASADO EN EQUIPOS KNX..... | 9 |

| | |
|--|------------------|
| 1.6.9. GESTIÓN DE ILUMINACIÓN MEDIANTE PROTOCOLO DE GESTIÓN KNX EN ALMACÉN DE ALIMENTACIÓN..... | 9 |
| <u>1.6.10. DIDACTIC EQUIPMENT DEVELOPED WITH EMBEDDED SYSTEMS FOR THE LEARNING OF ENGINEERING.</u> | <u>10</u> |
| 1.6.11. ADAPTIVE CONTROL IMPLEMENTED WITH EMBEDDED SYSTEMS | 10 |
| <u>CAPÍTULO 2: FUNDAMENTOS TEÓRICOS SOBRE LA AUTOMATIZACIÓN DE VIVIENDAS</u> | <u>11</u> |
| 2.1. HISTORIA DE LA DOMÓTICA | 11 |
| 2.2. ARQUITECTURA Y ELEMENTOS DE UN SISTEMA DOMÓTICO | 12 |
| 2.3. COMPONENTES DE UN SISTEMA DOMÓTICO. | 12 |
| 2.4. SENSOR..... | 13 |
| 2.5. ACTUADOR | 13 |
| 2.6. CONTROLADOR..... | 14 |
| 2.7. PASARELA RESIDENCIAL..... | 15 |
| 2.8. FUNCIONALIDAD | 15 |
| 2.9. ARQUITECTURA DEL SISTEMA DE CONTROL DOMÓTICO | 17 |
| 2.9.1. SISTEMAS CENTRALIZADOS | 17 |
| 2.9.2. SISTEMAS DESCENTRALIZADOS | 18 |
| 2.9.3. SISTEMAS MIXTOS (HÍBRIDOS)..... | 19 |
| 2.10. FIRMWARE..... | 19 |
| 2.11. PROBLEMAS USUALES DEL FIRMWARE | 20 |
| 2.12. EL MICROPROCESADOR..... | 20 |
| 2.13. MICROCONTROLADORES | 21 |
| 2.14. DIFERENCIA ENTRE MICROCONTROLADORES Y MICROPROCESADORES..... | 24 |
| 2.14.1. TECNOLOGÍA INVOLUCRADA EN MICROPROCESADOR Y MICROCONTROLADOR .. | 24 |
| 2.14.2. ARQUITECTURA DE MICROPROCESADOR Y MICROCONTROLADOR..... | 24 |
| 2.14.3. MEMORIA DE DATOS EN MICROPROCESADOR Y MICROCONTROLADOR | 25 |
| 2.14.4. APLICACIONES DE MICROPROCESADOR Y MICROCONTROLADOR | 25 |
| 2.14.5 ALMACENAMIENTO EN MICROPROCESADOR Y MICROCONTROLADOR | 25 |
| 2.15. PROTOCOLOS DE COMUNICACIÓN..... | 26 |
| 2.16. PROTOCOLO KNX | 26 |
| 2.17. ¿POR QUÉ USAR ESTE PROTOCOLO DE COMUNICACIÓN? | 28 |
| 2.18. APLICACIONES EN LAS CUALES SE UTILIZA EL PROTOCOLO DE COMUNICACIÓN KNX EN EL SECTOR TERCIARIO. | 29 |
| 2.19. APLICACIONES EN LAS CUALES SE UTILIZA EL PROTOCOLO DE COMUNICACIÓN KNX EN VIVIENDAS Y EDIFICIOS RESIDENCIALES..... | 32 |

| | |
|--|----|
| 2.19.1. CONTROL DE ILUMINACIÓN | 32 |
| 2.19.2. CONTROL DE PERSIANAS | 32 |
| 2.19.3. SISTEMA DE VENTILACIÓN Y CALEFACCIÓN | 32 |
| 2.19.4. CONTROL DE AUDIO Y VIDEO | 32 |
| 2.19.5. OPERACIÓN Y VISUALIZACIÓN | 32 |
| 2.19.6. SEGURIDAD | 32 |

CAPÍTULO 3: ESTUDIO Y DISEÑO DE UNA VIVIENDA CONTROLADA POR EL PROTOCOLO KNX

| | |
|--|----|
| 3.1 DISEÑO DE LA VIVIENDA PARA IMPLEMENTACIÓN | 34 |
| 3.2 TIPO DE CONTROLADOR USADO | 35 |
| 3.2.1. CARACTERÍSTICAS PRINCIPALES DEL PIC 16F877 | 35 |
| 3.2.2. DIAGRAMA DE PINS | 36 |
| 3.3. DISEÑO DEL PLANO ESQUEMÁTICO PARA LA IMPLEMENTACIÓN | 37 |
| 3.3.1. DISEÑO DE LA TARJETA MÁSTER | 37 |
| 3.3.2. PROGRAMACIÓN DE LA TARJETA MÁSTER | 39 |
| 3.3.3. DISEÑO ESQUEMÁTICO DE LA TARJETA ESCLAVA1 | 41 |
| 3.3.4. PROGRAMACIÓN DE LA TARJETA ESCLAVA 1 | 42 |
| 3.3.5. DISEÑO DE LA TARJETA ESCLAVA 2 | 44 |
| 3.3.6. PROGRAMACIÓN DE LA TARJETA ESCLAVA 2 | 45 |
| 3.4. DISEÑO PCB DE LA TARJETA MASTER | 47 |
| 3.5. DISEÑO PCB DE LA TARJETA SLAVE1 | 48 |
| 3.6. DISEÑO PCB DE LA TARJETA SLAVE2 | 49 |
| 3.7. CONEXIONES AL RELÉ DE ESTADO SOLIDO | 50 |

CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.....

| | |
|-----------------------------|----|
| 4.1. CONCLUSIONES: | 52 |
| 4.2. RECOMENDACIONES: | 52 |

ANEXOS

| | |
|---------------|----|
| ANEXO 1 | 54 |
| ANEXO 2 | 55 |
| ANEXO 3 | 56 |

BIBLIOGRAFÍA.....

Índice de Figuras

Capítulo 2

| | |
|---|----|
| Figura2.1: componentes de un sistema domotico..... | 27 |
| Figura2.2: Sensores varios..... | 28 |
| Figura2.3:Actuadores varios..... | 28 |
| Figura 2.4:Controladores..... | 29 |
| Figura 2.5: Arquitectura Inteligente..... | 32 |
| Figura 2.6: Arquitectura centralizada..... | 33 |
| Figura 2.7: arquitectura descentralizada..... | 34 |
| Figura 2.8: arq. Centralizada..... | 34 |
| Figura 2.9:Integracion de circuitos de un microcontrolador..... | 38 |
| Figura 2.10: KNX logo..... | 43 |
| Figura 2.11: aplicaciones básicas de KNX..... | 44 |
| Figura 2.12:botoneras Knx..... | 45 |
| Figura 2.13: Control de persianas KNX..... | 46 |
| Figura 2.14: detector de techo KNX..... | 46 |
| Figura 2.15: Aplicaciones que ofrece KNX..... | 48 |

Capítulo 3

| | |
|--|----|
| Figura3.1: Diseño de vivienda..... | 49 |
| Figura3.2: Diseño de recamara..... | 50 |
| Figura3.3: Diagrama del pic 16F877..... | 51 |
| Figura3.4: Tarjeta Master..... | 52 |
| Figura3.5: Tarjeta Master sensor..... | 53 |
| Figura3.6: Tarjeta Master LCD..... | 53 |
| Figura3.7: Tarjeta Master Programación 1..... | 54 |
| Figura3.8: Tarjeta Master Programación 2 | 55 |
| Figura3.9: Tarjeta Esclava 1..... | 56 |
| Figura3.10: Tarjeta Esclava 1 interruptores..... | 56 |
| Figura3.11: Tarjeta Esclava 1Programación 1..... | 57 |

| | |
|---|----|
| Figura3.12: Tarjeta Esclava 1 Programación 2..... | 58 |
| Figura3.13: Tarjeta Esclava 2 Diseño..... | 59 |
| Figura3.14: Tarjeta Esclava 2 Programación 1..... | 60 |
| Figura3.15: Tarjeta Esclava 2 Programación 2..... | 61 |
| Figura3.16: Tarjeta Master PCB..... | 62 |
| Figura3.17: Tarjeta Master PCB Buttom..... | 63 |
| Figura3.18: Tarjeta Slave1 PCB..... | 64 |
| Figura3.19: Tarjeta Slave1 PCB Buttom..... | 65 |
| Figura3.20: Tarjeta Slave2 PCB..... | 65 |
| Figura3.21: Tarjeta Slave2 PCB Buttom..... | 66 |
| Figura3.22: Relé de Estado solido..... | 67 |

Índice de Tablas

Capítulo 2

| | |
|---|----|
| Tabla2.1: Familias de los microcontroladores más comunes..... | 39 |
| Tabla2.2 Microprocesador vs microcontrolador..... | 41 |

Resumen

Al aparecer los dispositivos inteligentes como los ordenadores y los teléfonos celulares se fueron ideando una meta la cual era la automatización de un hogar, a lo cual se refiere a encendido y apagado de dispositivos de instalaciones eléctricas y electrónicas de forma centralizada o remota.

El presente trabajo de titulación tiene como finalidad presentar una aplicación domótica mediante el protocolo KNX, el cual explica su elevada relevancia, especialmente en la era tecnológica en la que nos encontramos.

Para la investigación se eligió el protocolo de comunicación KNX debido a la practicidad de comunicación, por ser compatible con arduinos, raspberries y pics, además de ~~y por~~ la confiabilidad de sus elementos como los switches de luz.

Los resultados de este trabajo de investigación corroboran las ventajas de la inversión y la confiabilidad del tipo de sistema en las viviendas de actualidad.

Palabras clave: DOMOTICA, KNX, ARDUINO, RASPBERRY, SWITCHES, PROTOCOLO DE COMUNICACIÓN

Abstract

When smart devices such as computers and cell phones appeared, a goal was developed, which was the automation of a home, which refers to the switching on and off of electrical and electronic installation devices in a centralized or remote way.

The purpose of the present titration work is to present a demotic application through the KNX protocol, which explains its high relevance, especially in the technological era in which we find ourselves.

For the research, the KNX communication protocol was chosen due to the practicality of communication, being compatible with arduinos, raspberries and pics, as well as the reliability of its elements such as light switches.

The results of this research work corroborate the advantages of investment and the reliability of the type of system in current housing.

Keywords: DOMOTICA, KNX, ARDUINO, RASPBERRY, SWITCHES, COMMUNICATION PROTOCOL

Capítulo 1: Descripción del trabajo de titulación

1.1. Introducción.

En la época que actual vivimos, es inevitable pensar sobre las aplicaciones que brindan las casas o edificios inteligentes, como son: el control de luces, temperatura, control de persianas, etc. El mundo electrónico brinda confort y seguridad en la vida diaria de los individuos, llegando a controlar inclusive las viviendas.

EL control domotico ayuda también en la parte económica para el usuario de vivienda, o cuando se lo utiliza en un edificio empresarial donde trabajan varias personas, ya que, por ejemplo, se puede programar para que haya luz en cierta zona del edificio cuando un sensor detecte movimiento así esta luz no estará prendida todo el día, ahorrando así la empresa gastos innecesarios que pueden ser perjudiciales para la misma.

La inversión económica que se realiza a través del protocolo propuesto, cuenta con grandes beneficios, debido a que si es utilizado a nivel de vivienda o en un edificio empresarial, la programación del encendido de luces, en base a sensores de movimiento, permitirá su ahorro y evitará el despilfarro innecesario, además que se contribuirá con la protección del medio ambiente.

Para esta investigación se aplicó el protocolo de comunicación KNX, el cual es muy utilizado en viviendas, debido a su entorno amigable con el usuario; además presenta versatilidad que puede ser utilizada en cualquier tipo de control de vivienda y su delay es casi indetectable; cabe indicarse que este tipo de protocolo puede ser vinculado con una gran variedad de dispositivos de diferentes marcas que trabajan con mismo protocolo KNX, por tal motivo, en los resultados del presente trabajo de titulación, se observarán los beneficios del uso y forma de funcionamiento.

1.2. Justificación.

El estudio del protocolo KNX aplicado a la domótica, presenta beneficios para los usuarios debido al tipo de tecnología, la misma que ofrece comodidad y seguridad para los dueños de hogares inteligentes, además del ahorro de energía, y por ende su contribución para la protección del medio ambiente.

La propuesta de este trabajo radica en la aplicación de control de luces mediante el protocolo KNX para posteriormente explicar la importancia de este tipo de control en la era tecnológica en la que actualmente nos encontramos.

Al finalizar este trabajo se podrán observar los beneficios y ventajas que ofrece esta aplicación, la misma que brinda confort y seguridad a los usuarios en sus viviendas, con una inversión que perdurará por largo tiempo.

1.3. Planteamiento del problema.

La finalidad de la inmótica es optimizar el funcionamiento de distintas obras que la arquitectura moderna pone al servicio de los individuos. No todas las construcciones cuentan con un sistema de seguridad adecuado, situación que impide el funcionamiento eficaz de las mismas; por ese motivo, la falta de la aplicación de automatización ocasiona que las construcciones sean inseguras, cuando se trata de: detección de robos, incendios y todo aquello que el avance de la tecnología puede ofrecer. El presente estudio se enfocará en la aplicación de un protocolo especializado en comunicación de componentes que utiliza la domótica, para demostrar la importancia y necesidad de su uso en las viviendas actuales, atribuyéndoles seguridad y confort.

1.4. Objetivos de la investigación

1.4.1 Objetivo General.

Aplicar el protocolo de comunicación KNX para controlar las luces de una vivienda unifamiliar.

1.4.2. Objetivos específicos.

- Investigar la importancia de un sistema domótico.
- Puntualizar los beneficios del protocolo de comunicación KNX.
- Realizar la simulación del circuito en proteus.
- Elaborar tarjetas electrónicas mediante microcontroladores para la elaboración de un prototipo para el control de luces de una vivienda.
- Diseñar programas en lenguaje C para micro controladores utilizando el protocolo de comunicación KNX

1.5. Metodología de la investigación

En el presente trabajo de titulación se utilizará el método científico porque se buscamos la obtención de información teórica mediante el uso de instrumentos, este caso tarjetas electrónicas y el programa pic c, que se utilizará para compilar el programa, corroborar que refleje errores y trabaje sin ninguna novedad; también se utilizarán tarjeta para demostrar que la herramienta sirve en una vivienda real demostrando los beneficios en cualquier tipo de vivienda donde se requiera este sistema para comodidad.

1.6. Estados del Arte.

1.6.1. Sistema domótico para una casa inteligente

Desde mediados del siglo XX se han organizado varias exhibiciones donde se presentan ideas de la apariencia y funcionamiento de las viviendas en un futuro lejano. Las personas han podido imaginar cómo obtener mayor comodidad durante la estancia en sus hogares, así como la facilidad en la realización de las diferentes tareas domésticas, etc. Después de la aparición de dispositivos electrónicos inteligentes, como el ordenador, ha surgido el concepto de la automatización del hogar: la domótica [1], el cual se refiere a la automatización y control (encendido, apagado, apertura, cierre y regulación) de aparatos y sistemas de instalaciones eléctricas y electrotécnicas (iluminación, climatización, persianas y toldos, puertas y ventanas motorizadas, el riego, etc.) de forma centralizada y/o remota. El objetivo principal del uso de la domótica es el aumento del confort, el ahorro energético y la seguridad del hogar.

Autor: Marcos Peirotn

1.6.2. Diseño e implementación de un prototipo de vivienda domótica basado en las plataformas arduino y android

Este artículo presenta el desarrollo de un prototipo de casa inteligente de bajo costo como una experiencia en área de la domótica. El objetivo principal es reportar el proceso y la metodología que podrían ser usados por otros grupos que tengan como propósito un proyecto similar. Este reporte incluye fallos y aciertos clave que se tuvieron durante los diferentes procesos: planeación, diseño y ejecución del prototipo

Autores: Camila Sánchez, Alejandra Mesa, Carolina Manrique, Herbert Calderón, Luis Cobo, Rubén Dorado, Camilo Mejía.

1.6.3. Diseño de un módulo para el control de iluminación de un hardware domótico a través de un teléfono celular en un ambiente distribuido

El proyecto reportado en este artículo estuvo orientado al desarrollo de una aplicación para un dispositivo celular (midlet) capaz de interactuar con un hardware domótico que simula a escala el ambiente de una casa; el objetivo es controlar el encendido y apagado de las luces que lo conforman, manteniendo las características de un sistema distribuido. Para lograr este objetivo, se diseñará un sistema de Servlets siguiendo patrones de programación de J2EE como controladores frontales, despachadores y acceso a datos, junto con un servidor y cliente RMI. La aplicación midlet utilizará RMS para guardar y consultar datos y un computador llamado controlador domótico enviará por el puerto serial una orden de apagado o encendido al hardware domótico para que sea ejecutada.

Autores: Carlos Fernando Varela, Julián Augusto Bojacá Medina, Angela Viviana Ramírez Acevedo.

1.6.4. Estudio de prefactibilidad para implementar sistemas domóticos en las casa del sector de La Española en la ciudad de Bogotá para potenciar el bienestar humano y el medio ambiente.

El presente proyecto está orientado a realizar un estudio de prefactibilidad de un sistema domótico o sistema de automatizaciones de hogares que permita controlar varios dispositivos básicos. Está dirigido principalmente para los habitantes de las viviendas del barrio la Española, localidad de Engativá, en la ciudad de Bogotá. Se entiende por domótica el conjunto de sistemas capaces de automatizar una vivienda, aportando servicios de gestión energética, seguridad, bienestar y comunicación, cuyo control goza de cierta ubicuidad, desde dentro y fuera del hogar. Para el desarrollo del proyecto, se contemplaron varios puntos como: la realización de encuestas conformadas por 7 preguntas claves donde se puede determinar los aspectos socioeconómicos de los clientes, el grado de conocimiento sobre domótica y la aplicación que puede tener en sus hogares. Luego se realiza la tabulación de los resultados obtenidos para su respectivo análisis. Posteriormente, se

presenta el estudio de mercado donde nos muestra la información del sector y de la competencia para obtener mayores elementos de juicio en la toma de decisiones para realizar el análisis financiero del proyecto.

Autores: Cifuentes Castillo, Edgar Esteban; Román Rodríguez, Wilson Fredy; Romero Zipa, Sandra Mayerly.

1.6.5. Diseño e implementación de una casa domótica para perros domésticos usando tecnología raspberry pi y arduino con control a través de interfaz web.

El presente proyecto de titulación “Diseño e implementación de una casa domótica para perros domésticos usando tecnología Raspberry pi y Arduino con control a través de interfaz web” tiene como objetivo brindar un cuidado ideal para los perros, facilitando la labor a los dueños, debido a que por sus arduas labores cotidianas se les imposibilita darle la atención que ellos necesitan. Para el desarrollo del mismo se contó con un sistema de acceso para el ingreso del can, junto con el dispensador de comida anexo a un grabador de voz que facilita la integración del perro con su nuevo hogar, haciendo del mismo un lugar placentero y acogedor, también posee un sistema de ventilación para mantener un clima idóneo, un sistema de iluminación y un sistema de vigilancia para que el dueño monitoree el estado y las necesidades del can, todo ello es controlado mediante una tarjeta Raspberry Pi 3 junto a una tarjeta Arduino Mega, las cuales adquieren los datos obtenidos de los sensores ubicados en la casa y proceden a ejecutar acciones específicas que el usuario selecciona mediante la interfaz web. Teniendo como resultado final una herramienta útil para ayudar al cuidado del perro, gracias a la domótica empleada en ello.

Autores: Bastidas Martínez, Leonel Andrés, Rodríguez Sares, Kerly Andrea

1.6.6. Gestión domótica de una casa unifamiliar basada en Arduino

La realización de este proyecto busca la gestión domótica de una casa unifamiliar basada en Arduino, para ello se estudiarán las distintas características de los sistemas domóticos, su manejo mediante controladores, sensores y actuadores y la comunicación entre estos. Además se desarrollarán los sistemas domóticos para una casa y se profundizará en el control e integración de todos los sistemas mediante Arduino, buscando una gestión eficaz y que se pueda implementar físicamente

Autor : Rodriguez Diego Miguel

1.6.7. Diseño e implementación de un sistema de domótica basado en la tecnología smart bus KNX para el control de iluminación, audio y seguridad, mediante un enlace web Apps

El presente proyecto tiene como finalidad diseñar e implementar un sistema de domótica usando la tecnología SMART BUS KNX, aplicado en una casa ubicada en el norte de la ciudad de Quito, para controlar la iluminación, audio y seguridad en forma local mediante mandos manuales llamados paneles de control o DLPs y en forma remota mediante el diseño de un aplicativo web para equipos Smart como tablets, celulares y computadores. El proyecto abarca varios tipos de luces entre las cuales se tiene del tipo LED, del tipo LED RGB, del tipo incandescente que permite controlar la luminosidad y escenas diseñadas de las distintas áreas. En audio se dispone de parlantes empotrados en el techo, un amplificador y un módulo de control de audio que permite enlazar cualquier tipo de periféricos con salida o entrada de audio. En seguridad se dispone de control de accesos, sensores de movimiento o presencia, sensor de humo, botón de pánico, circuito cerrado de video vigilancia y alerta sonora enlazada al módulo de audio para control de emergencias. Finalmente el aplicativo web instalado en los equipos Smart dispone de todos los controles para las escenas, de iluminación, audio y seguridad enlazados.

Autores: Sánchez Cunalata, Daniel Fabricio

1.6.8. Desarrollo e implementación de un sistema basado en equipos KNX

El objetivo del presente trabajo es el montaje y la programación de un sistema domótico con tecnología KNX. Inicialmente se realiza una breve introducción a la domótica. Se explica que es la domótica, se relata brevemente la historia de la domótica, se enumeran sus objetivos principales y se habla de los diferentes componentes de un sistema domótico. Tras haber introducido el tema general del trabajo, se realiza una introducción a la tecnología KNX. Ésta es la tecnología que se va a utilizar para llevar a cabo el trabajo. Además, también se explica como funciona dicha tecnología, como transmite y estructura los datos y sus principales ventajas e inconvenientes. A continuación, se introduce la herramienta de programación utilizada, ETS5, detallando los aspectos más importantes de este software y su funcionamiento. Después, se presentan los diferentes componentes utilizados en el sistema domótico y como se realizará el conexionado. Por último, una vez montado el sistema domótico, se lleva a cabo la programación y se comprueba el correcto funcionamiento del sistema.

Autor: Maza Alcañiz, Iván

1.6.9. Gestión de iluminación mediante protocolo de gestión KNX en almacén de alimentación

La redacción del proyecto tiene como objetivo la instalación de un sistema inmótico que permita la gestión de iluminación, tanto a nivel de contactores como de sistema DALI del establecimiento. El proyecto realizado se centra en la instalación del sistema inmótico del edificio y en la instalación eléctrica en las zonas que lo requieren, tanto a nivel de cableado como de canalizaciones.

Autor: BENLLOCH VERDEGUER, MARINA

1.6.10. Didactic Equipment Developed with Embedded Systems for the Learning of Engineering.

In this written is presented the development of a didactic plant, of low cost, in which is possible to do practices of classic and advanced control. The variable to control is the luminous intensity, this equipment will strengthen the students' practice knowledge of careers related to the industrial control area. The implementation counts with an electric circuit, which has a smart sensor, an embedded system, and as actuator elements LED diodes, a monitoring and control interface has been development using the Matlab Simulink tool. Finally, there is the physical part where the control of illumination is made. The equipment were tested, applying a PID type controller, what check the factibility level of the using of the plant as an instrument to proof control algorithms. This system will give to universities the opportunity of provider of a low cost device by their automatic control laboratories.

Autores: Nino Vega, Pablo Parra, Daniel Martillo

1.6.11. Adaptive Control Implemented with Embedded Systems

In this research paper, we describe the implementation of an adaptive control algorithm for programmed gains, applied to a liquid level signal. The driver has been embedded in an ARDUINO business card. A test plant was used to perform the control tests. Two different models were obtained for the plant and the performance analysis of the two models was carried out, compared to the performance of the testing plant. Finally, the diagrams and results obtained in the simulations as well as in the real plant are shown.

Autores: Nino Vega, Pablo Parra, Luis Córdova, Joselyne Andramuño, Víctor Navarrete

Capítulo 2: Fundamentos teóricos sobre la automatización de viviendas

2.1. Historia de la Domótica

En el año 1977, países como Estados Unidos y Japón fueron los que iniciaron la implementación de inmuebles inteligentes bajo el predominio de factores tecnológicos y económicos. Con la llegada de las tecnologías en comunicaciones y el surgimiento de la nueva generación de los conmutadores telefónicos llamados de multiservicio o PABX, se produjeron los primeros avances en áreas de edificios inteligentes. Estos novedosos sistemas de comunicación admitían la transmisión de datos numéricos y la conversación telefónica simultáneamente, produciendo que este último servicio fuese monopolizado hasta el año 1984, por la compañía de comunicaciones AT&T. (Carlos Daniel Ramirez, 2011)

El proyecto llamado “Smart House” fue dirigido por la NAHB (“National Association of Home Builders”), asociación integrada por constructores de casas unifamiliares que crearon una fundación para impulsar el desarrollo de la casa inteligente. (Carlos Daniel Ramirez, 2011)

En 1987, la idea fue desarrollada en Japón con la finalidad de lograr espacios que proporcionen ambientes que brinden comodidad y seguridad, y de esa manera hacerlos competitivos en el mercado de la construcción.

Hablar de automatización en el hogar, se refiere a una nueva tecnología; en el momento de convertir una casa normal a una inteligente, se puede comprobar que se permite a los usuarios mayor comodidad. Inicialmente el control de los dispositivos se ejecutaba por red eléctrica, para luego cambiar y mejorar, hasta llegar a los emisores y receptores más avanzados, los cuales reciben la señal y la transforman en una acción determinada.

La palabra domótica tiene su origen de la unión de dos palabras del latín: *Domo* que significa “casa” y *Tica* que significa “funciona por si sola”; en 1998, el término fue incluido en la enciclopedia Larousse, la cual definió a la palabra

como “vivienda que integra todos los automatismos en materia de seguridad gestión de la energía ya comunicaciones”.

La domótica busca la perfecta armonía, con máxima utilidad y mínima intervención por parte del usuario. Ofrece la posibilidad de gestionar un sistema inteligente mediante la modificación local o remota de los parámetros de la instalación a partir de sistemas centralizados, descentralizados, distribuidos o una combinación de estos, que recogen información del entorno a través de sensores, la información es procesada y enviada como ordenes a los actuadores. Así, un sistema domótico, adquiere cierto grado de inteligencia que le permite brindar a los hogares, nuevas y beneficiosas aplicaciones que pueden agruparse en servicios básicos para el hogar, como: gestión energética o ahorro energético, confort, seguridad y comunicación. (Valencia, 2016)

2.2. Arquitectura y elementos de un sistema domótico

Los elementos y arquitectura de un sistema domótico involucran un estudio y análisis planificado y la rápida detección a tiempo de fallas en los medios de transmisión del sistema domótico en la vivienda. (Wilber, 2016)

2.3. Componentes de un sistema domótico.

Un sistema domótico se conforma básicamente por tres grupos de elementos: sensores, controladores y actuadores, los que ejecutan procesos puntuales en el proceso de automatización. En la siguiente figura mencionamos los componentes domoticos.

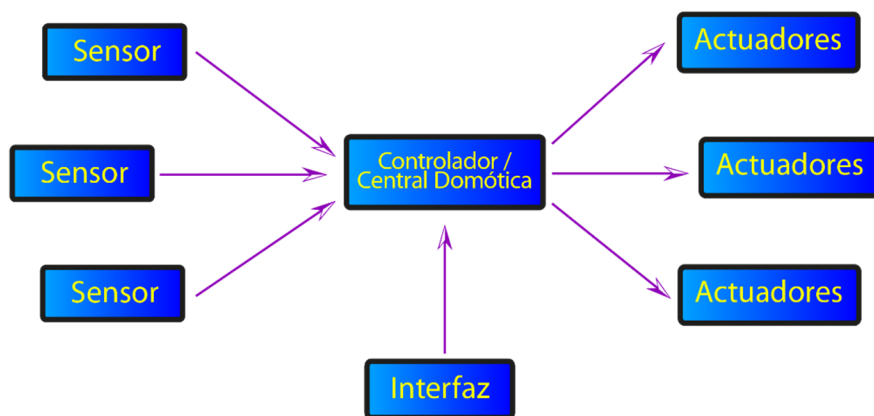


Figura2.1: Componentes de un sistema domótico
Fuente: (Domoticaudem,2017)

2.4. Sensor

Los sensores son elementos que reciben información de variables físicas, las cuales monitorean constantemente el entorno como luz y temperatura, para así realizar un evento que será procesado por el controlador con el cual se está comunicando. Los sensores pueden identificar variables de valores máximo y mínimo, los cuales se pueden visualizar en la hoja del fabricante. (Aguirre Quiroz, 2011)



Figura2.2: Sensores varios
Fuente:(rockwell automation,2017)

Tipos de sensores:

- Sensor de proximidad.
- Sensor de gas.
- Sensor de incendios.
- Sensor de humedad.
- Sensores de presencia /volumétricos y perimetrales.
- Sensor de iluminación
- Sensor de lluvia.
- Sensor de temperatura, etc. (Wilber, 2016)

2.5. Actuador

Son los dispositivos que reciben la información de los sistemas y por lo tanto, los activan o desactivan según las variables que mide el sensor; los actuadores son dispositivos de salida que permiten ejecutar acciones como: encendido/apagado, apertura/ cierre etc. (Aguirre Quiroz, 2011)



Figura2.3:Actuadores varios
Fuente: LITEM

2.6. Controlador

El sistema de control no es más que el operador que enlaza el sensor con el actuador; su función radica en recibir la señal del sensor y por medio de la programación del sistema, enviar un mensaje ya sea de inhibición o activación.

En el controlador se encuentra la inteligencia del sistema y donde se procesa la información. (Aguirre Quiroz, 2011)



Figura 2.4:Controladores
Fuente: Siemens

2.7. Pasarela residencial

Los dispositivos se enlazan y usan en la automatización de la vivienda, haciendo una interfaz común de todos ellos hacia las redes externas, permitiendo el control remoto de todos los dispositivos de la vivienda. (Wilber, 2016)

2.8. Funcionalidad

La Domótica como tal, presenta una gran variedad de servicios y aplicaciones que pueden ser integradas en una instalación, y entre las que se pueden mencionar están:

Seguridad

- Seguridad perimetral e interior, con sistema de alertas telefónico, celular y SMS.
- Grabación y almacenamiento de imágenes del circuitos cerrados de TV.
- Control de accesos.
- Sistemas de seguridad eléctrica (disyuntores, etc.).
- Alarmas médicas.
- Simulación de presencia.
- Alarmas Técnicas: agua, fuego, gas, humo. (Aguirre Quiroz, 2011)

Control y gestión energética

- Control de demanda de consumos generales (Bombas y Motores).
- Control de eficiencia de equipos de aire acondicionado, zonificados.
- Control de iluminación automatizada exterior e interior.
- Control del suministro de energía (fuentes alternativas).
- Desconexión selectiva de cargas eléctricas.
- Activación/desactivación de la iluminación en función del nivel de umbral de iluminación.
- Lectura remota de contadores.
- Control de riego.
- Control de paneles solares.

Confort

- Activación de rutinas y funciones, mediante control horario, ocurrencia de eventos o manual.
- Automatización de tareas rutinarias.
- Monitoreo y gestión en forma local y remota.
- Programación de estilos de vida.
- Estación Meteorológica.
- Simplificación de la vida diaria.

Comunicaciones

- Comunicación con el exterior para aviso de incidencias (teléfono, e-mail, SMS).
- Integración de portero automático en la telefonía interior.
- Control de la vivienda desde el teléfono interior, PDA o emisor infrarrojo.
 - Control de la vivienda desde el exterior, mediante telefonía fija y móvil, Internet, etc. 1.2.3.5 Automatización.
- Gestión de electrodomésticos en función de la sobrecarga.
- Activación/desactivación de la iluminación por detección de presencia.
- Centralización de apagado encendido general de la iluminación.
- Control de accesos.
- Acciones preventivas automáticas (cierre de válvulas de agua, gas).

(Aguirre Quiroz, 2011)

2.9. Arquitectura del sistema de control domótico



Figura 2.5: Arquitectura Inteligente
Fuente: Mundo Digital

Existen tres tipos de arquitecturas:

- Sistemas Centralizados
- Sistemas descentralizados
- Sistemas Distribuidos

2.9.1. Sistemas Centralizados

Poseen un único nodo el cual recibe la información de las entradas para después procesarlo y enviarlo a las salidas de los dispositivos que están unidos a una central que ejecuta funciones de mando y control. (Wilber, 2016)

Domótica: su aplicación en la empresa y en la sociedad globalizada

Arquitectura Domótica Centralizada



Figura 2.6: Arquitectura centralizada
Fuente: DLNA

2.9.2. Sistemas descentralizados

Los elementos de red actúan de forma independiente y poseen la misma línea de comunicación; disponen de funciones de control y mando y necesita de un protocolo de comunicaciones para que todos los elementos hagan una acción coordinada. (Wilber, 2016)

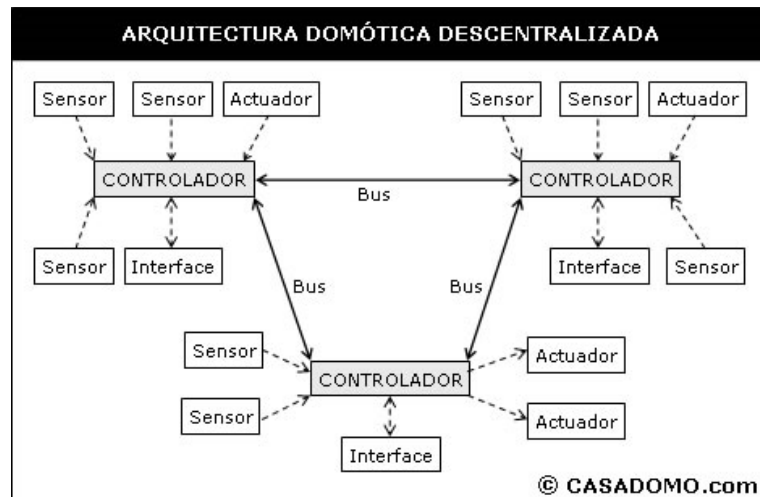


Figura 2.7: arquitectura descentralizada
Fuente: Wordpress

2.9.3. Sistemas Mixtos (híbridos)

Se enlazan las arquitecturas centralizadas y distribuidas; su trabajo se encuentra localizado en cada uno de los nodos de control, y cada nodo tiene acceso a una serie limitada de elementos de red que controla las zonas de manera distribuida; también de manera centralizada se requiere de un protocolo de comunicaciones para que la coordinación sea ordenada. (Wilber, 2016)

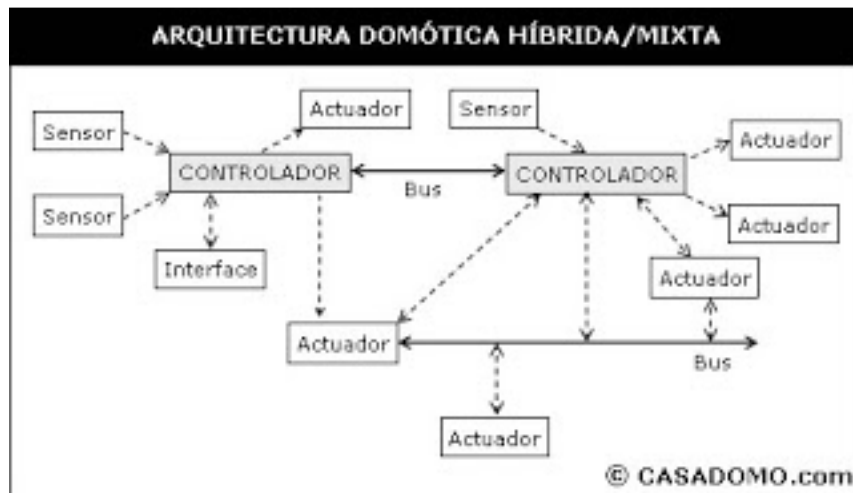


Figura 2.8: Arq. Centralizada
Fuente: Domótica Blogspot

2.10. Firmware

El firmware se refiere al conjunto de instrucciones de un programa informático que se encuentra registrado en una memoria ROM, flash o similar. Estas instrucciones fijan una lógica primaria que ejerce el control de los circuitos de alguna clase de artefacto. (Muñoz, 2016)

El firmware forma parte del hardware, porque se encuentra integrado a la electrónica, pero también es considerado parte del software, al estar desarrollado bajo un lenguaje de programación. Se podría decir que funciona como el nexo entre las instrucciones que llegan al dispositivo desde el exterior y sus diversas partes electrónicas.

Cualquier firmware debe cumplir tres funciones claras: En primer lugar, otorgar al sistema las rutinas fundamentales de funcionamiento y respuesta

con respecto a las peticiones usuales que recibe y debe satisfacer al usuario. (Muñoz, 2016)

En segundo plano, establecer una interfaz para que, de esta forma, se pueda realizar la configuración del sistema de forma rápida, y finalmente, controlar y gestionar el arranque del sistema del dispositivo, así como la iniciación.

En la actualidad se pueden encontrar firmware en multitud de dispositivos, como las impresoras, monitores o periféricos. (Muñoz, 2016)

2.11. Problemas usuales del Firmware

Uno de los grandes inconvenientes del firmware es la falta de actualización automática, lo que puede provocar ciertos fallos en su funcionamiento habitual. Por ello constantemente se lanzan nuevas versiones de firmware, que deben ser actualizados de forma manual por el usuario. (Muñoz, 2016)

2.12. El Microprocesador

El microprocesador es un chip integrado basado en silicio con solo una unidad de procesamiento central. Es el corazón de un sistema informático que está diseñado para realizar muchas tareas que involucran datos. Los microprocesadores no tienen RAM, ROM, IO pins, temporizadores y otros periféricos en el chip. Deben agregarse externamente para que sean funcionales. Consiste en la ALU que maneja todas las operaciones aritméticas y lógicas; la Unidad de control que gestiona y maneja el flujo de instrucciones en todo el sistema; y Register Array, que almacena los datos de la memoria para un acceso rápido. Están diseñados para aplicaciones de propósito general, como operaciones lógicas en sistemas informáticos. En términos simples, es una CPU completamente funcional en un solo circuito integrado que es utilizado por un sistema informático para hacer su trabajo.

2.13. Microcontroladores

Cuando se trata de automatizar una vivienda, ya sea por luminarias o alguna otra forma de control, es necesario usar un microcontrolador el mismo que es un circuito integrado, es el componente principal de una aplicación embebida. Se lo considera como una pequeña computadora que incluye sistemas para controlar elementos de entrada/salida. También incluye a un procesador y por supuesto memoria que puede guardar el programa y sus variables (flash y RAM). Funciona como una mini PC. Su función es la de automatizar procesos y procesar información. (E-Marmolejo., 2017)

ELEMENTOS INTERNOS EN UN MICROCONTROLADOR

- **Procesador o Microprocesador.** Un procesador incluye al menos tres elementos: ALU, unidad de control y registros.
- **ALU.-** También conocida como Unidad Aritmética y Lógica, la cual está compuesta por los circuitos electrónicos digitales del tipo combinatorios (compuertas, sumadores, multiplicadores), cuya principal función es realizar operaciones que pueden estar divididas en los siguientes tres tipos:
 1. **Lógicas:** Comprende como las operaciones básicas de las compuertas lógicas, como la suma lógica (OR), multiplicación lógica (AND), diferencia lógica (XOR) y negación (NOT). Una operación lógica sólo puede tener como entradas y como salidas una respuesta lógica (0 o 1). Esto dependiendo de los niveles de voltajes de una señal digital. (E-Marmolejo., 2017)
 2. **Aritméticas:** Las operaciones aritméticas son la suma, resta, multiplicación y división. Dependiendo del procesador (8, 16, 32 o 64 bits) será la rapidez con la que se pueden hacer dichas operaciones.
 3. **Misceláneas:** En estas operaciones caen todas las demás operaciones como la transferencia de bits (<< >>). (E-Marmolejo., 2017)
- **Unidad de control:** La unidad de control es el conjunto de sistemas digitales secuenciales (aquellos que tienen memoria) que permiten distribuir la lógica de las señales.

- **Registros:** Los registros son las memorias principales de los procesadores, ya que funcionan a la misma velocidad que el procesador a diferencia de otras memorias un tanto más lentas (como la RAM, FLASH o la CACHE). Los registros están contruidos por Flip-Flops, que son circuitos digitales secuenciales. (E-Marmolejo., 2017)
- **Periféricos:** Son los circuitos digitales que permiten una interacción desde el mundo “exterior” al microcontrolador. Su función es la de poder habilitar o deshabilitar las salidas digitales, leer sensores analógicos, comunicación con terminales digitales o sacar señales analógicas de una conversión digital.
- **Puertos de entrada/salida paralelos:** Los puertos están relacionados al tamaño del procesador, es decir, que un puerto de 8 bits es porque el procesador es de 8 bits. Un procesador de 64 bits, tiene la capacidad de tener un puerto de 64 bits. (E-Marmolejo., 2017)
- **Puertos seriales:** Permiten transformar la información digital paralela (bytes de información) en tramas que se pueden transferir por una o varias líneas de comunicación. Entre los que existen están siguientes: puerto serial, i2c, SPI, USB, CAN, etc.
- **Periféricos analógicos:** Son los que convierten señales analógicas a digitales (ADC) o señales digitales a analógicas (DAC) o comparadores analógicos.
- **Memoria:** La memoria está dividida en tres: La memoria para el programa (FLASH), la memoria para los datos o variables del programa (RAM) y la memoria para configuraciones o no volátil (EEPROM). (E-Marmolejo., 2017)

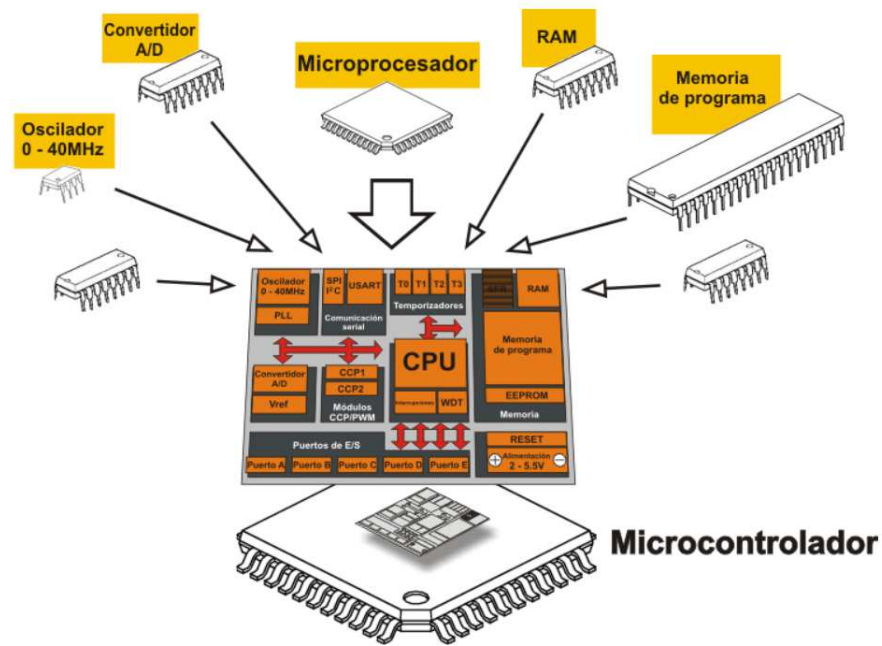


Figura 2.9: Integración de circuitos de un microcontrolador
Fuente: Taller Electrónica

El uso de microcontroladores se ha universalizado, pero a diferencia de lo que ocurre con los microprocesadores, los microcontroladores de 8 bits, debido a que son extremadamente económicos y sencillos de programar, se continúan fabricando y utilizando en infinidad de aplicaciones electrónicas, por lo que se los puede considerar un estándar en aquellas aplicaciones que no requieren gran capacidad de proceso de información, es decir, que no necesiten, algún tipo de reconocimiento de imagen, entorno gráfico propio o procesos estadísticos complejos. También se emplean microcontroladores de 8 bits en subsistemas que se encargan de labores muy especializadas dentro de sistemas electrónicos más complejos. (Hernandez, 2014)

Se los encuentran ordenados por familias que cuentan con características y herramientas de programación comunes, de manera que el diseñador de circuitos electrónicos de control es capaz de escalar dentro de los distintos microcontroladores de la familia, desde el más sencillo, hasta el más completo o complejo, sin necesidad de cambiar entorno o filosofía de programación. (Hernandez, 2014)

A continuación se realizó una tabla sobre las familias de los microcontroladores.

| <u>Empresa</u> | <u>Familia</u> | <u>Estructura</u> |
|------------------|--|---|
| <u>Atmel</u> | <u>AT89xxx</u> <u>TS87xxx</u> <u>AVR</u> <u>AT91xxx</u> | <u>CISC 8 bits 8051</u> <u>CISC 8 bits</u> <u>RISC 8 bits</u> <u>Risc 16 bits ARM 7/9</u> |
| <u>Freescale</u> | <u>HC05</u> <u>HC08</u> <u>HC11</u> <u>HC12</u> <u>HCS12</u> <u>HC16</u> <u>58800</u> <u>68K</u> <u>ColdFire</u> <u>MAC7100</u> | <u>CISC 8 bits 6800</u> <u>CISC 8 bits 6809</u> <u>CISC 8 bits 6809</u> <u>CISC 16 bits</u> <u>CISC 16 bits</u> <u>CISC 16 bits</u> <u>CISC 16 bits</u> <u>CISC 32 bits 68000</u> <u>CISC 32 bits</u> <u>RISC 32 bits ARM7</u> |
| <u>Microchip</u> | <u>PIC10,12,14,16,17,18</u> <u>dsPIC</u> | <u>RISC 8 bits</u> <u>RISC 16 bits</u> |

Tabla2.1: Familias de los microcontroladores más comunes

Elaborado por: Autor

2.14. Diferencia entre microcontroladores y microprocesadores

2.14.1. Tecnología involucrada en microprocesador y microcontrolador

El microprocesador es un chip de silicio multipropósito programable, que es el componente más crítico dentro de un sistema informático. Se lo podría comparar como si fuese un corazón del sistema informático que consta de ALU (Unidad Lógica Aritmética), unidad de control, decodificadores de instrucciones y matriz de registro. El microcontrolador, además es el corazón del sistema integrado.

2.14.2. Arquitectura de microprocesador y microcontrolador

El microprocesador es solo un circuito integrado sin RAM, ROM o pines de entrada / salida. Básicamente se refiere a la unidad de procesamiento central del sistema informático que capta, interpreta y ejecuta los comandos que se le pasan. Incorpora las funciones de una CPU en un solo circuito integrado. Los microcontroladores, por otro lado, son dispositivos más potentes que contienen los circuitos del microprocesador y tienen RAM, IO y procesador, todo en un solo chip. (Jhon, 2018)

2.14.3. Memoria de datos en microprocesador y microcontrolador

La memoria de datos es una parte del PIC que contiene Registros de funciones especiales y Registros de fines generales. Almacena datos temporalmente y mantiene resultados intermedios. Los microprocesadores ejecutan varias instrucciones que se almacenan en la memoria y envían los resultados a la salida. Los microcontroladores contienen una o más CPU junto con la RAM y otros periféricos. La CPU obtiene las instrucciones de la memoria y ejecuta los resultados. (Jhon, 2018)

2.14.4. Aplicaciones de microprocesador y microcontrolador

Los microprocesadores son un dispositivo de almacenamiento masivo con un solo chip y están integrados en varias aplicaciones, como: control de especificación, control del semáforo, control de temperatura, instrumentos de prueba, sistema de monitoreo en tiempo real y mucho más. Los microcontroladores se utilizan principalmente en circuitos eléctricos y electrónicos y en dispositivos controlados automáticamente, como instrumentos médicos de alta gama, sistemas de control de motores de automóviles, cargadores solares, máquinas de juegos, control de semáforos, dispositivos de control industrial, etc.

2.14.5 Almacenamiento en microprocesador y microcontrolador

Los microprocesadores se basan en la arquitectura de Von Neumann (también conocida como modelo de Von Neumann y arquitectura de Princeton), en la que la unidad de control obtiene las instrucciones asignando señales de control al hardware y decodificando. El objetivo es almacenar las instrucciones en la memoria junto con los datos sobre los cuales operan las instrucciones. Los microcontroladores, por otro lado, se basan en la arquitectura de Harvard donde las instrucciones y los datos del programa se almacenan por separado. (Jhon, 2018)

A continuación se realizó una tabla sobre la diferencia entre los microprocesadores y los microcontroladores.

| Microprocesador | Microcontrolador |
|---|--|
| No tiene memoria RAM O ROM Timers u otros periférico en el chip | Tiene un CPU además de MEMORIAS RAM y ROM. |
| Es el corazón de un sistema computarizado. | Es el cerebro de un sistema computarizado. |
| Está diseñado para propósitos generales como aplicaciones que manejan innumerables datos. | Son utilizados para aplicaciones específicas del sistema. |
| Es complejo, costoso y con un amplio número de instrucciones para procesar. | Es simple, barato y con menos número de instrucciones para procesar. |

Tabla2.2 Microprocesador vs Microcontrolador
Elaborado por: Autor

2.15. Protocolos de Comunicación

En la domótica no existe un protocolo de comunicación, existen varios protocolos muy extendidos, robustos y fiables, entre ellos, los más utilizados por fabricantes e instaladores son: el protocolo KNX, el protocolo X10 o el ZigBee. (Monter, 2017)

2.16. Protocolo KNX

Se trata de un estándar de protocolo de comunicaciones de red, basado en OSI, para edificios inteligentes; este estándar es la convergencia de tres estándares previos: el European Home Systems Protocol (EHS), el European Installation Bus (EIB) y el BatiBus. El estándar KNX está gestionado por la asociación homónima KNX y el texto de la especificación KNX se adquiere previo pago de una tarifa.

Los tres estándares previos, intentaron desarrollar sus mercados separadamente, pero, en 1997, los tres consorcios decidieron unir sus fuerzas para desarrollar el mercado del hogar inteligente, acordando crear una norma industrial común para poder proponerla como norma internacional. En 2002 fue presentada la especificación KNX, logrando entrar lentamente en el mercado, a pesar de ser un sistema bastante robusto y fiable. (Monter, 2017)

KNX define varios medios de comunicación física, tales como: el cableado de par trenzado, la comunicación por red eléctrica, la comunicación mediante radiofrecuencia o el ethernet. La especificación KNX consta de cuatro grupos de elementos que son los siguientes:

- * **Actuadores:** Son los elementos del sistema que se conectan físicamente con los elementos a controlar en un edificio, como por ejemplo: las luces, los motores, las válvulas o electroválvulas; estos actuadores se encargan de traducir las instrucciones recibidas del mundo KNX al mundo físico. (Monter, 2017)

- * **Sensores:** Son los elementos del sistema que recogen los datos del entorno o interpretan las órdenes del usuario, como por ejemplo: los pulsadores, los detectores de movimiento o el termostato.

- * **Pasarelas:** Las pasarelas enlazan otros sistemas con otros protocolos de comunicación a KNX; a estos equipos se les permiten interactuar con proyectores, otros sistemas inteligentes o comunicarse en remoto con nuestro sistema. (Monter, 2017)

- * **Acopladores:** Los acopladores realizan una separación física dentro del bus, consiguiendo agrupar los dispositivos en un segmento de características determinadas para la cantidad de equipos, ubicaciones físicas o funciones determinadas, para conectarlos con otro segmento y dar mayor eficacia en el envío de datagramas a través del bus o alcanzar mayores distancias (Repetidores). (Monter, 2017)

El futuro próximo de esta especificación, parece ser que pasa por la encapsulación de las comunicaciones dentro del protocolo IP, con la especificación KNX/IP, y de esta manera se puedan conectar los dispositivos compatibles con el estándar KNX a las redes que la mayoría de viviendas y oficinas tienen ya instaladas, es decir, las redes TCP/IP con cableado UTP. (Monter, 2017)



Figura 2.10: KNX logo
Fuente: Blogseas

2.17. ¿Por qué usar este protocolo de comunicación?

KNX se diferencia de los demás protocolos por ser abierto y por ende es utilizado por varios fabricantes.

Para la domótica o la inmótica se necesita un sistema que sea independiente de fabricantes y este protocolo de comunicación cuenta con más de 24 años de experiencia, tiempo durante el cual ha desarrollado procesadores como EIB, EHS, y BatiBUS.

Por medio del trenzado que KNX ofrece, al conectar todos los dispositivos que han sido utilizados, se intercambia la comunicación. Los dispositivos conectados al bus pueden ser sensores así como actuadores, utilizados para el control del equipamiento de gestión de edificios en todas las aplicaciones posibles: iluminación, persianas / contraventanas, sistemas de seguridad, gestión energética, calefacción, sistemas de ventilación y aire acondicionado, sistemas de supervisión y señalización, interfaces a servicios y sistemas de control de edificios, control remoto, medición, audio / video, control de bienes de gama blanca, etc. (KNX, 2018)

Beneficios que ofrece KNX

El ahorro energético es el resultado del bajo coste operacional. La iluminación y la calefacción sólo se encenderán cuando sea necesario dependiendo de los perfiles seleccionados y/o de la presencia en la habitación, de esta forma se ahorra energía y dinero. Cabe indicar que la iluminación puede ser controlada automáticamente según la intensidad de luz natural, manteniendo

un nivel mínimo de luminosidad en cada lugar de trabajo y reduciendo de esta manera el consumo energético. (KNX, 2018)

Es relevante reconocer que se puede enlazar todos los dispositivos de control sencillamente a través de un bus, lo que reduce considerable el tiempo de diseño e instalación.

Existe una única herramienta que es independiente de la aplicación y del fabricante llamada «Engineering Tool Software» (ETS) la que permite el diseño, la implementación y la configuración de la instalación que posean productos certificados KNX. (KNX, 2018)

Una instalación KNX puede ser fácilmente adaptada y ampliada a nuevas aplicaciones. Los nuevos componentes se pueden conectar directamente a la instalación bus existente. (KNX, 2018)

2.18. Aplicaciones en las cuales se utiliza el protocolo de comunicación KNX en el sector terciario.

Al ser KNX un estándar en lo que se refiere a la automatización de viviendas y edificios, puede utilizarse en cualquiera de las siguientes aplicaciones:



Figura 2.11: aplicaciones básicas de KNX
Elaborado por: Autor

Al decir control energético nos referimos que el protocolo KNX realiza las siguientes operaciones:

- Monitorización de picos de demanda.
- Detección de corrientes.
- Monitorización de redes.
- Control de cargas.
- Medición.
- Contador de impulsos de energía.
- Almacenamiento de datos.
- Visualización (KNX, 2018)

En el control de luminarias se maneja lo siguiente:

- Encendido/apagado y regulación.
- Iluminación automática.
- Control de iluminación constante.
- Control por tiempos.
- Escenas de iluminación.
- Pasarela con DALI (KNX, 2018)



Figura 2.12:Botoneras Knx
Fuente: Simon Electric

Al usar el protocolo KNX, el confort es una de las prioridades, por lo que esta comunicación está especializada en las siguientes operaciones:

- Control centralizado o por grupos.
- Posición por defecto.
- Seguimiento solar.
- Programas automáticos.
- Control en función de la climatología.
- Protección contra viento y lluvia.
- Modos de seguridad.
- Pasarela SMI (KNX, 2018)



Figura 2.13: Control de persianas KNX
Fuente: GIRA

Otras de las prioridades de este protocolo es la seguridad que se refleja en los siguientes aspectos:

- Intrusión.
- Humo/Fuego.
- Fallos técnicos.
- Control de Acceso.
- Tecnologías preventivas.
- Simulación de presencia.
- Monitorización de fallos.
- Supervisión del Bus (KNX, 2018)



Figura 2.14: detector de techo KNX
Fuente: Vendomática

Con el control remoto se puede contar con la opción de monitorear cada edificio mientras el individuo no se encuentre presente, pues a este sistema se cuenta con las siguientes opciones:

- Funciones Lógicas.
- Funciones Temporizadas.
- Supervisión del Sistema.
- Acceso a Internet.

- Control Remoto.
- Programación Remota.
- Mensajes (KNX, 2018)

2.19. Aplicaciones en las cuales se utiliza el protocolo de comunicación KNX en viviendas y edificios residenciales

El sistema fue desarrollado para controlar las viviendas, donde el usuario se sienta cómodo y a gusto dentro de su hogar y tener la potestad de monitorear la misma sin ningún esfuerzo.

Lo que permite controlar este protocolo de comunicaciones en el sector residencial es:

2.19.1. Control de Iluminación

- Control centralizado de la iluminación en la vivienda y en el jardín.
- Elección de diferentes escenas de iluminación o regulaciones individuales.

2.19.2. Control de persianas

- Control de persianas, toldos y contraventanas según las condiciones de viento, luminosidad y/o lluvia.
- Control de persianas y contraventanas según programación horaria y semanal.

2.19.3. Sistema de ventilación y calefacción

- Control de calefacción automática y optimizada de acuerdo a los usos o necesidades de sus ocupantes.
- Las ventanas serán abiertas o cerradas de acuerdo a lo que sea requerido.
- El sistema de ventilación se activa según haya presencia o no de personas en la habitación.

2.19.4. Control de audio y video

- Control remoto de música desde cualquier punto de la vivienda.
- Control remoto para todas y cada una de las habitaciones.

2.19.5. Operación y visualización

- Visualización y operación de todos los sistemas de la vivienda a través de panel (táctiles).
- Fácil visualización e integración de los sistemas de audio y cámaras de monitorización.

2.19.6. Seguridad

- Informes de ventanas y puertas abiertas o forzadas, aviso ante peligro de robo, emisión de humos, etc.

- Cámaras de monitorización para la entrada.
- Disuasión del peligro potencial de intrusión activando la totalidad de la iluminación («modo pánico»).
- Simulación de vivienda ocupada a través del control temporizado de iluminación y persianas. (KNX, 2018)

| | | | | |
|---|--|--|--|--|
|  Iluminación |  Calentación |  Climatización |  Ventilación |  Persianas, toldos |
|  Aplic. interior |  Aplic. exterior |  |  Medición |  Efic. energética |
|  Alarma intrusión |  Alarma incendio |  Sanitarias |  Electrodomésticos |  Audio/Vídeo |

Figura 2.15: Aplicaciones que ofrece KNX
Fuente: Domótica con KNX

Capítulo 3: Estudio y Diseño de una vivienda controlada por el protocolo KNX

3.1 Diseño de la vivienda para implementación

Este trabajo de investigación alcanzará su validez cuando en primera instancia diseñe una vivienda acorde a lo que se ha mencionado que este equipo puede controlar, motivo por el que se elaboró el dibujo de una vivienda básica, para una sola persona con cuatro ambientes básicos (sala, cocina, baño y recámara), lo que permitirá demostrar que este control funciona en los ambientes indicados, los mismos que son propios de cualquier tipo de residencia.

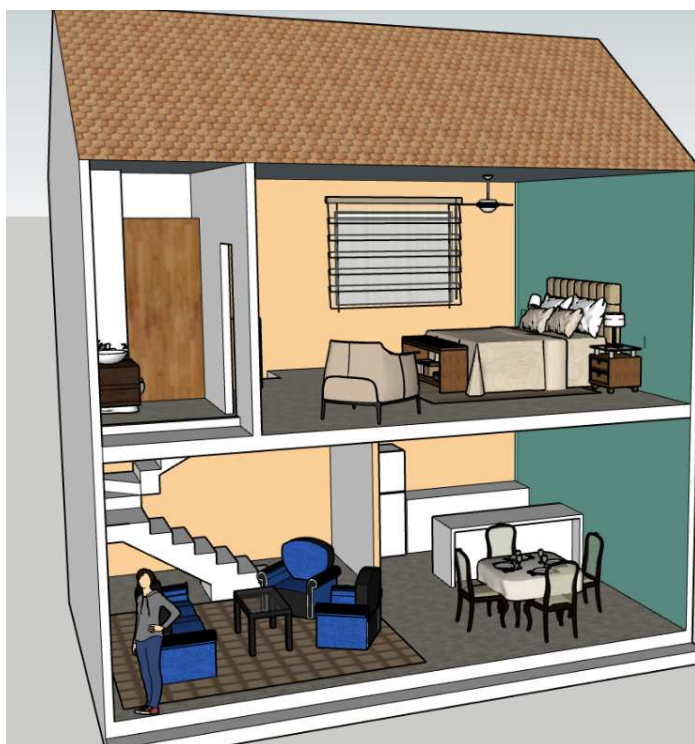


Figura 3.1: Diseño de vivienda
Elaborado por: Autor

Como se muestra en la figura 3.1, se colocó un foco de 110V en cada ambiente de la casa, es decir, uno en la recámara, en el baño, en la sala y en la cocina, además se instaló un sensor de temperatura en la cocina, por ser uno de los lugares con mayor riesgo en una vivienda, y de esta manera se

podrá monitorear y conocer si la temperatura es la deseada para este ambiente de la casa.

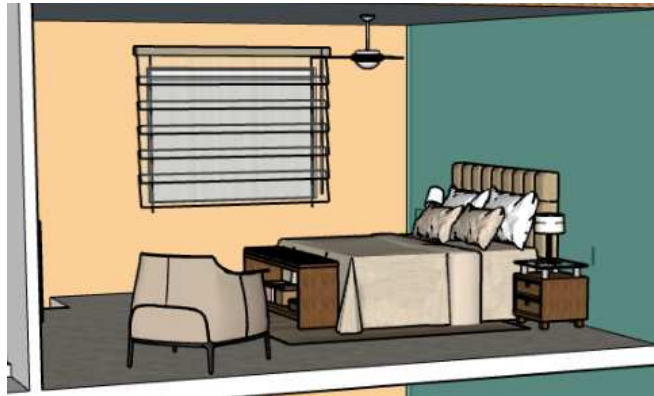


Figura3.2: Diseño de recámara
Elaborado por: Autor

En la recámara, además de la iluminación se colocará un sensor infrarrojo que permitirá detectar a la persona al momento de ingresar, porque se lo ubicará a la entrada y así el usuario no tendrá que buscar el interruptor en los momentos de oscuridad; de esta manera se puede corroborar la eficiencia del sistema porque su comunicación al ser inmediata no permite la pérdida de tiempo en hacer llegar la señal al controlador, y por ende se encenderá la luminaria sin necesidad de un interruptor.

3.2 Tipo de controlador usado

Para el presente trabajo se utilizó un tipo de microcontrolador muy especial: el pic 16F877, que fue escogido por su gran capacidad de memoria y velocidad de trabajo, siendo capaz de soportar la velocidad del protocolo de comunicación del KNX.

3.2.1. Características principales del pic 16F877

- CPU RISC de alta performance.
- Set de 35 instrucciones.
- Todas las instrucciones son de un ciclo salvo aquellas que incluyen saltos que son de 2 ciclos.

- Velocidad de Trabajo: DC - 20 MHz clock input DC - 200 ns ciclo de instrucción.
- Hasta 8K x 14 words de FLASH Program Memory, Hasta 368 x 8 bytes de Data Memory (RAM) Hasta 256 x 8 bytes de EEPROM Data Memory.
- Manejo de Interrupciones.
- Stack de hardware de 8 niveles.
- Modo de direccionamiento directo, indirecto y relativo. • Power-on Reset (POR) .
- Power-up Timer y Oscillator Startup Timer.
- Watchdog Timer con el reloj RC interno para mejor seguridad.
- Protección de código programable.
- Programación serial via 2 pines,
- In-Circuit Debugging via 2 pines
- Amplio rango de voltaje de trabajo: 2.0V a 5.5V

3.2.2. Diagrama de Pins

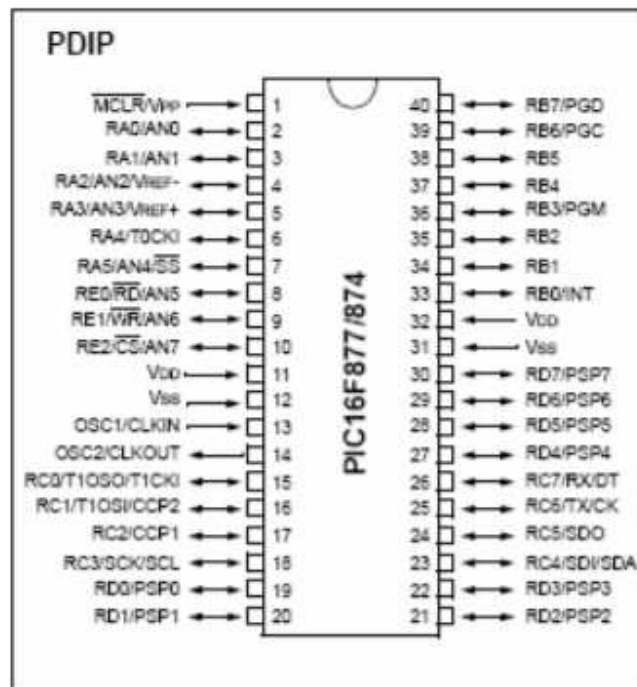


Figura3.3: Diagrama del pic 16F877

Fuente: www.fing.edu.uy.pdf

3.3. Diseño del plano esquemático para la implementación

Para implementar el proyecto se necesitará una guía, por lo que se presenta un plano o esquema, elaborado con el programa Proteus y en el que se podrá observar cómo será la organización de cada placa electrónica, para que funcione la vivienda sin ningún inconveniente.

3.3.1. Diseño de la Tarjeta Máster

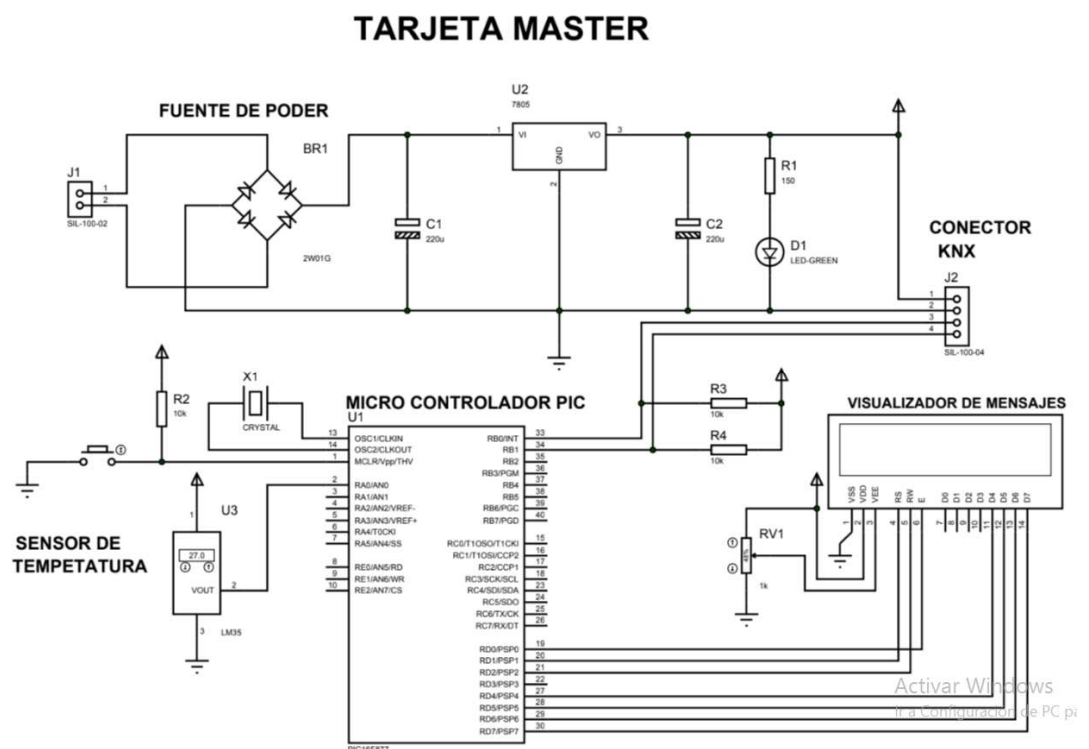


Figura 3.4: Tarjeta Máster
Elaborado por: Autor

Como se puede observar en la Figura 3.4 se puede identificar primero, como en todo dispositivo electrónico es necesario colocar una fuente de alimentación, que brinde energía al pic para que a su vez pueda cargar el programa. En este caso es la tarjeta máster, la misma que genera las órdenes a las otras tarjetas que serán mencionadas más adelante.

Esta tarjeta máster ordena a las demás, el momento en que deben actuar y en qué circunstancias o parámetros.

Además de generar órdenes a las demás tarjetas electrónicas, la tarjeta master está programada para recibir información de un sensor de temperatura el cual está colocado en un lugar de la vivienda que permita la medición de temperatura de forma constante.

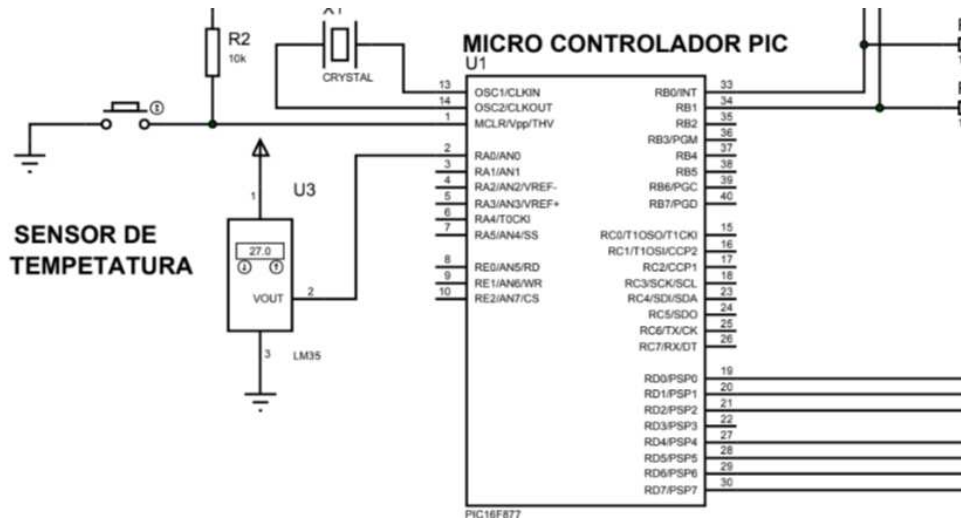


Figura3.5: Tarjeta Máster sensor
Elaborado por: Autor

Al medir la temperatura, la información será transmitida al microcontrolador desde donde se podrá visualizar en un LCD, el mismo que registrará todo el tiempo la temperatura del dormitorio y así el usuario conocerá si existen o si se mantiene en una misma temperatura.

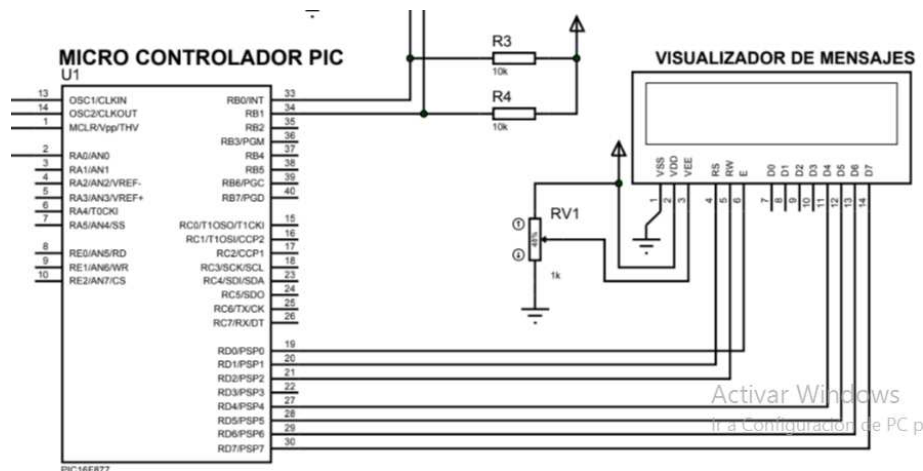


Figura3.6: Tarjeta Máster LCD
Elaborado por: Autor

3.3.2. Programación de la Tarjeta Máster

La programación que se utiliza para el protocolo y los microcontroladores es el lenguaje c, la misma que se ejecutó en el programa PIC C por la facilidad de su uso y la eficacia que este programa provee.

```
#include <16f877a.h>
#fuses xt,nowdt
#use delay(clock=4000000)
#include <KnxDevice.h>
#include <Cli.h> // command line interpreter lib available at https://github.com/franckmarini/Cli
Cli cli = Cli(Serial);
String traces;
KnxComObject objList[] =
{
  KnxComObject(0x0001, KNX_DPT_1_001 /* 1.001 B1 DPT_Switch */ , COM_OBJ_LOGIC_IN) , // Switch actuator command
};
byte resetEvt, newTgEvt, newStateEvt, ackEvt;
e_TpUartTxAck ackVal;

void TracesDisplay() { printf(traces); traces=""; }

void PrintTelegramInfo(KnxTelegram& tg) { traces = " => Info() :\n"; tg.Info(traces); TracesDisplay(); }

boolean Pulse400us(void)
{
  static word before = (word) micros();
  word now = (word) micros();

  if ((word)(now -before) > 400)
  {
    before = now;
    return true;
  }
  else return false;
}

boolean Pulse800us(void)
{
  static word before = (word) micros();
  word now = (word) micros();

  if ((word)(now -before) > 800)
  {
    before = now;
    return true;
  }
}
```

Figura3.7: Tarjeta Máster Programación 1
Elaborado por: Autor

En esta primera parte de la programación se declara el tipo de microcontrolador que se utilizará, siendo en este caso el PIC 16F877; posteriormente se declara como KNX device, y de esa manera se realiza el protocolo que lleva el mismo nombre; como todo dispositivo debe de contar con un botón de encendido y apagado que estará declarado en la programación; además se puede observar la línea de código para el funcionamiento del LCD.

```

{
  ackEvt++;
  ackVal = ack;
}

void Reset_Tests(void);          // Test Reset function
void Attach_Tests(void);        // Attach Tests
void Init_Tests(void);          // Test Init function
void Bus_Monitoring(void);      // Test Bus Monitoring mode
void Normal_Rx(void);           // Test addressed telegrams reception
void Normal_Rx_ResetEvt(void);  // Test Reset Event reception
void Normal_Rx_StateEvt(void);  // Test State Event reception
void Normal_Tx_Val0(void);      // Test telegram transmission : Send boolean value 1 to valid address 0x0001 (switch actuator OFF)
void Normal_Tx_Val1(void);      // Test telegram transmission : Send boolean value 0 to valid address 0x0001 (switch actuator ON)
void Normal_Tx_NoAck(void);     // Test telegram transmission with NoAck (target address does not exist)
void Normal_Tx_Timeout(void);   // Test telegram transmission with no answer timeout (no answer got from the TPUART)
void Normal_Tx_Reset(void);     // Test telegram transmission with reset response
void All_Tests(void);

void main(){
  cli.RegisterCmd("reset",&Reset_Tests);
  cli.RegisterCmd("attach",&Attach_Tests);
  cli.RegisterCmd("init",&Init_Tests);
  cli.RegisterCmd("moni",&Bus_Monitoring);
  cli.RegisterCmd("rx",&Normal_Rx);
  cli.RegisterCmd("rxreset",&Normal_Rx_ResetEvt);
  cli.RegisterCmd("rxstate",&Normal_Rx_StateEvt);
  cli.RegisterCmd("tx0",&Normal_Tx_Val0);
  cli.RegisterCmd("tx1",&Normal_Tx_Val1);
  cli.RegisterCmd("txnoack",&Normal_Tx_NoAck);
  cli.RegisterCmd("txnoans",&Normal_Tx_Timeout);
  cli.RegisterCmd("txreset",&Normal_Tx_Reset);
  cli.RegisterCmd("all",&All_Tests);

  while(1){
    Knx.begin(Serial, P_ADDR(1,1,1)); // start a KnxDevice session with physical address 1.1.1 on Serial UART
    Knx.task();
  }
}
void knxEvents(byte index) {

```

Figura3.8: Tarjeta Máster Programación 2
Elaborado por: Autor

En esta parte de la programación se maneja la parte del sensor de temperatura para que lo que mide se registre en el micro y que lo muestre en el LCD, así cada cierto tiempo, el micro enviará la orden que el sensor vuelva a medir y así el micro resetea el LCD para que la nueva medición sea mostrada en el mismo. Adicional a este proceso, también se envían órdenes para las otras dos tarjetas electrónicas esclavas, las cuales están encargadas del funcionamiento de la iluminación.

3.3.3. Diseño esquemático de la tarjeta Esclava1

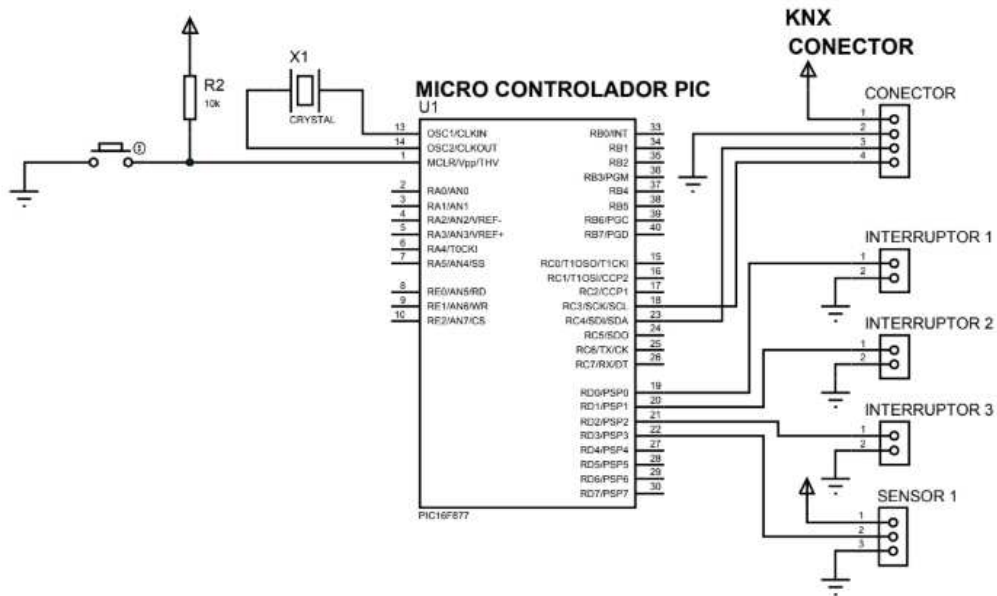


Figura3.9: Tarjeta Esclava 1
Elaborado por: Autor

A esta tarjeta se le han colocado tres interruptores y un sensor, y básicamente se encarga de dar la señal de encendido y apagado; mediante los interruptores esta señal va hacia el micro para que la señal se desplace hacia la otra tarjeta; también podría suscitarse que el sensor genere la señal que este recoge y viaje hacia la otra tarjeta para activar la luminaria.

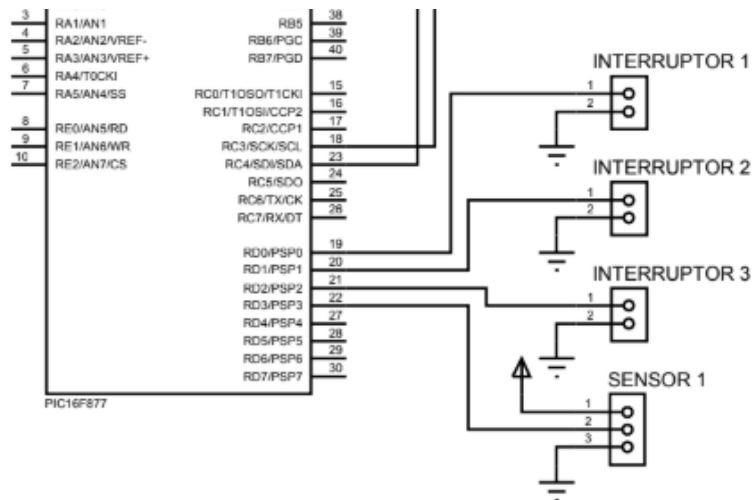


Figura3.10: Tarjeta Esclava 1 interruptores

3.3.4. Programación de la Tarjeta Esclava 1

```
#include <16f877a.h>
#fuses xt,nowdt
#use delay(clock=4000000)
#include <KnxDevice.h>
#include <Cli.h> // command line interpreter lib available at https://github.com/franckmarini/Cli
Cli cli = Cli(Serial);
String traces;
KnxComObject objList[] =
{
  KnxComObject(0x0001, KNX_DPT_1_001 /* 1.001 B1 DPT_Switch */ , COM_OBJ_LOGIC_IN) , // Switch actuator command
  KnxComObject(0x0002, KNX_DPT_1_001 /* 1.001 B1 DPT_Switch */ , COM_OBJ_LOGIC_IN) , // feedback status of the switched actuator
  KnxComObject(0x0003, KNX_DPT_1_001 /* 1.001 B1 DPT_Switch */ , COM_OBJ_LOGIC_IN) , // Push button command
};
byte resetEvt, newTgEvt, newStateEvt, ackEvt;
e_TpUartTxAck ackVal;

// WARNING : "KNXTPUART_DEBUG_INFO" and "KNXTPUART_DEBUG_ERROR" flags shall be set in order to get KnxTpUart traces
void TracesDisplay() { printf(traces); traces=""; }

void PrintTelegramInfo(KnxTelegram& tg) { traces = " => Info() :\n"; tg.Info(traces); TracesDisplay(); }

boolean Pulse400us(void)
{
  static word before = (word) micros();
  word now = (word) micros();

  if ((word)(now -before) > 400)
  {
    before = now;
    return true;
  }
  else return false;
}

boolean Pulse800us(void)
{
  static word before = (word) micros();
  word now = (word) micros();

  if ((word)(now -before) > 800)
```

Figura3.11: Tarjeta Esclava 1 Programación 1
Elaborado por: Autor

Con esta programación se elaboró un código para que el micro reconozca los interruptores y el sensor, y así enviar la información a la última tarjeta que se encarga de activar las luces, así como también indicarle a la tarjeta máster, cuales son las luces que están encendidas y cuales no, por lo que de esta manera, las tarjetas tienen un control general del sistema y de la vivienda en su totalidad.


```

{
byte return_val;
  printf(F("\n##### Attach tests #####"));

  {
    printf(F("### Attach with error KNX_TPUART_ERROR_NOT_INIT_STATE
(254)"));
    KnxComObject list[] = { KnxComObject(0x0000, KNX_DPT_1_001 /*
1.001 B1 DPT_Switch */ , COM_OBJ_LOGIC_IN) , };
    KnxTpUart tpuart(Serial1, 0x1234, NORMAL);
    return_val = tpuart.AttachComObjectsList(list, sizeof
(list)/sizeof(KnxComObject));
    printf(F("Attach return val = ")); printf(return_val);
  }
  TracesDisplay();
  {
    printf(F("\n### Attach an ordered list with com attributes
objects and no duplicate addr"));
    KnxComObject list[] =
    {
      /* index 0 */ KnxComObject(0x0000, KNX_DPT_1_001 /* 1.001 B1
DPT_Switch */ , COM_OBJ_LOGIC_IN) ,
      /* index 1 */ KnxComObject(0x0001, KNX_DPT_1_001 /* 1.001 B1
DPT_Switch */ , COM_OBJ_LOGIC_IN) ,
      /* index 2 */ KnxComObject(0x0002, KNX_DPT_1_001 /* 1.001 B1
DPT_Switch */ , COM_OBJ_LOGIC_IN) ,
      /* index 3 */ KnxComObject(0x0003, KNX_DPT_1_001 /* 1.001 B1
DPT_Switch */ , COM_OBJ_LOGIC_IN) ,
      /* index 4 */ KnxComObject(0x0004, KNX_DPT_1_001 /* 1.001 B1
DPT_Switch */ , COM_OBJ_LOGIC_IN) ,
    }
  }
}

```

Figura3.12: Tarjeta Esclava 1 Programación 2
Elaborado por: Autor

En esta parte de la programación se puede observar la identificación de los interruptores y sensores con las bibliotecas de KNX, logrando esa comunicación única que tiene el sistema, logrando de esta manera, que las señales sean más rápidas, convirtiéndose en una de las funciones más relevantes de toda la programación, por ser el corazón de la misma, pues sin él, el sistema no funcionaría.

3.3.5. Diseño de la Tarjeta Esclava 2

Esta es la tarjeta donde básicamente llegan todas las señales de salida, donde se prenden o apagan las luces y se demuestra que este sistema de comunicación es confiable y que puede ejecutar estas acciones con más rapidez que cualquier otro tipo de protocolo de comunicación.

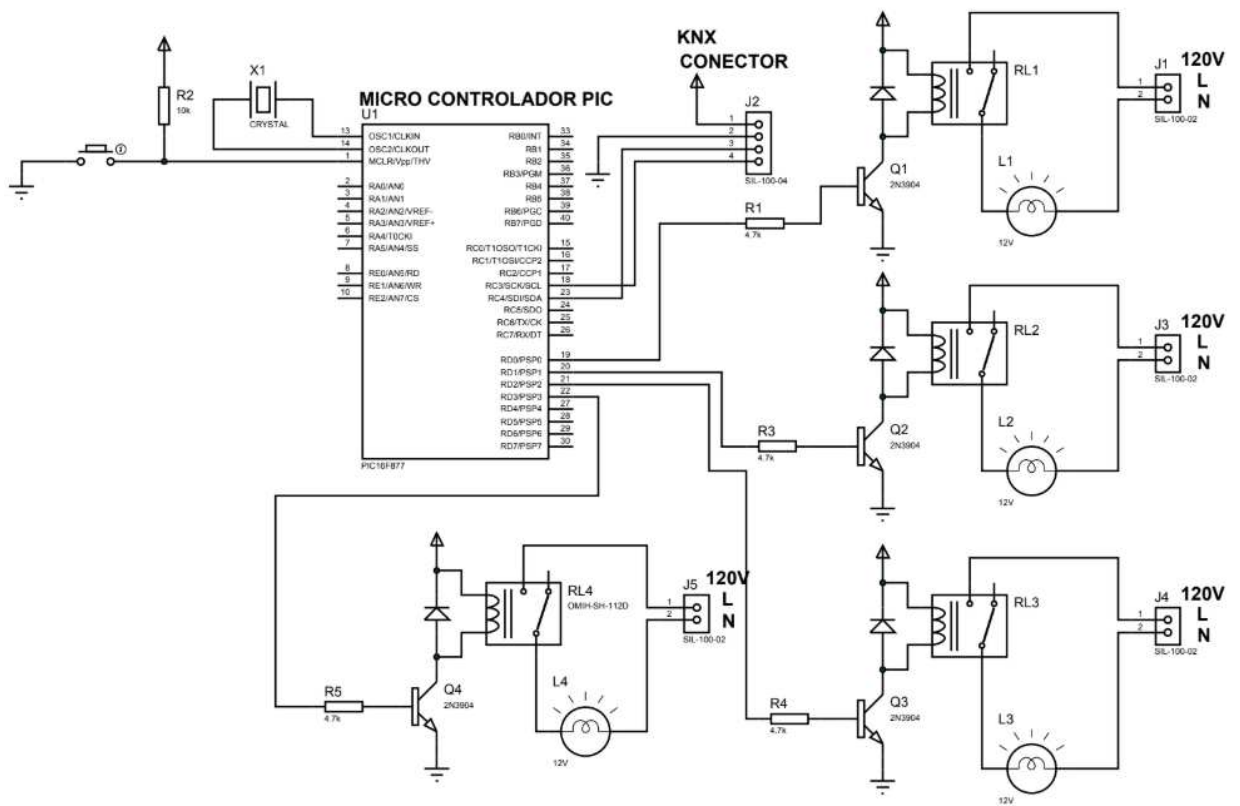


Figura 3.13: Tarjeta Esclava 2 Diseño
Elaborado por: Autor

En la figura 3.13 se pueden observar los focos de 110 V que serán conectados a un relé que verificará cuál de las señales están pasando por él para encender las luces, así como las que no lo hacen. Además por este medio se pueden identificar los leds que este relé tiene a su cargo.

3.3.6. Programación de la Tarjeta Esclava 2

```
#include <16f877a.h>
#fuses xt,nowdt
#use delay(clock=4000000)
#include <KnxDevice.h>
#include <Cli.h> // command line interpreter lib available at
https://github.com/franckmarini/Cli
Cli cli = Cli(Serial);
String traces;
KnxComObject objList[] =
{
  KnxComObject(0x0001, KNX_DPT_1_001 /* 1.001 B1 DPT_Switch */ ,
COM_OBJ_LOGIC_IN) , // Switch actuator command
};
byte resetEvt, newTgEvt, newStateEvt, ackEvt;
e_TpUartTxAck ackVal;

void TracesDisplay() { printf(traces); traces=""; }

void PrintTelegramInfo(KnxTelegram& tg) { traces = " => Info() :\n";
tg.Info(traces); TracesDisplay(); }

boolean Pulse400us(void)
{
static word before = (word) micros();
word now = (word) micros();

if ((word)(now -before) > 400)
```

Figura3.14: Tarjeta Esclava 2 Programación 1
Elaborado por: Autor

En esta programación se ejecutan los retoques finales para conocer cuándo la señal de las demás tarjetas lleguen a ella, así como cuál de las luces se deben encender o apagar, cuál es la que está monitoreada por el sensor y cuáles por los interruptores, además de toda esta información a la tarjeta maestra, para que la organice y se conozca todo lo que sucede en el sistema; este tipo de comunicación es básicamente de dos vías: la tarjeta maestra da las ordenes y las otras dos le responden con el estado de la operación.

```

}

void Reset_Tests(void);           // Test Reset function
void Attach_Tests(void);         // Attach Tests
void Init_Tests(void);           // Test Init function
void Bus_Monitoring(void);       // Test Bus Monitoring mode
void Normal_Rx(void);            // Test addressed telegrams reception
void Normal_Rx_ResetEvt(void);   // Test Reset Event reception
void Normal_Rx_StateEvt(void);   // Test State Event reception
void Normal_Tx_Val0(void);       // Test telegram transmission : Send
boolean value 1 to valid address 0x0001 (switch actuator OFF)
void Normal_Tx_Val1(void);       // Test telegram transmission : Send
boolean value 0 to valid address 0x0001 (switch actuator ON)
void Normal_Tx_NoAck(void);      // Test telegram transmission with
NoAck (target address does not exist)
void Normal_Tx_Timeout(void);    // Test telegram transmission with no
answer timeout (no answer got from the TPUART)
void Normal_Tx_Reset(void);      // Test telegram transmission with
reset response
void All_Tests(void);

void main(){
cli.RegisterCmd("reset",&Reset_Tests);
cli.RegisterCmd("attach",&Attach_Tests);
cli.RegisterCmd("init",&Init_Tests);
cli.RegisterCmd("moni",&Bus_Monitoring);

```

Figura3.15: Tarjeta Esclava 2 Programación 2
Elaborado por: Autor

3.4. Diseño PCB de la tarjeta master

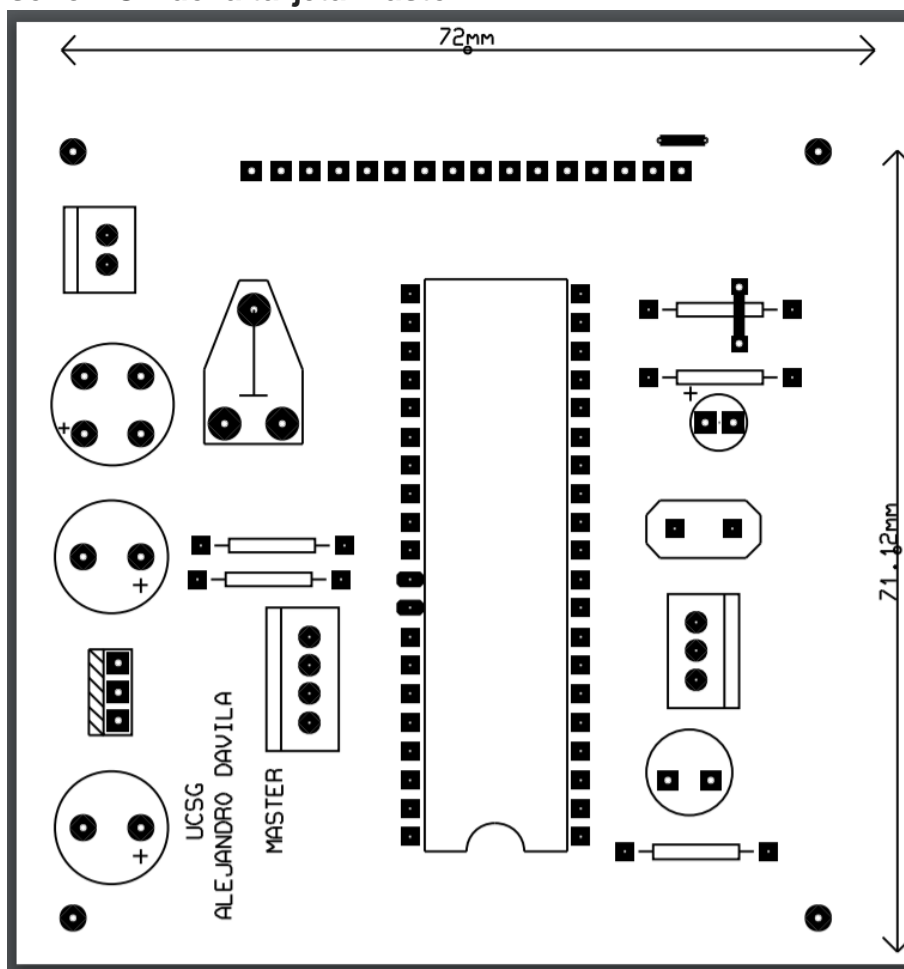


Figura3.16: Tarjeta Master PCB
Elaborado por: Autor

En este diagrama se puede observar como se conectarán los elementos que conforman la tarjeta electrónica máster, es decir, donde se ubicará el LCD, el sensor de temperatura etc.

Es muy importantes distribuir eficientemente los elementos de una tarjeta para ocupar el menor espacio posible, pues lo de mayor relevancia en la electrónica, lo importante es simplificar las cosas lo más que se pueda, sin que esta situación genere daño en el funcionamiento del sistema que se pretende ejecutar.

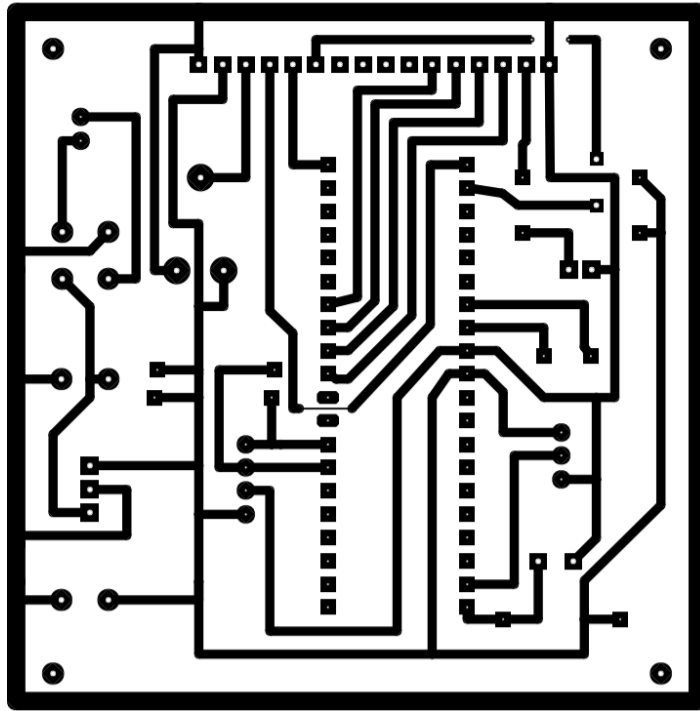


Figura3.17: Tarjeta Máster PCB Bottom
Elaborado por: Autor

3.5. Diseño PCB de la tarjeta slave1

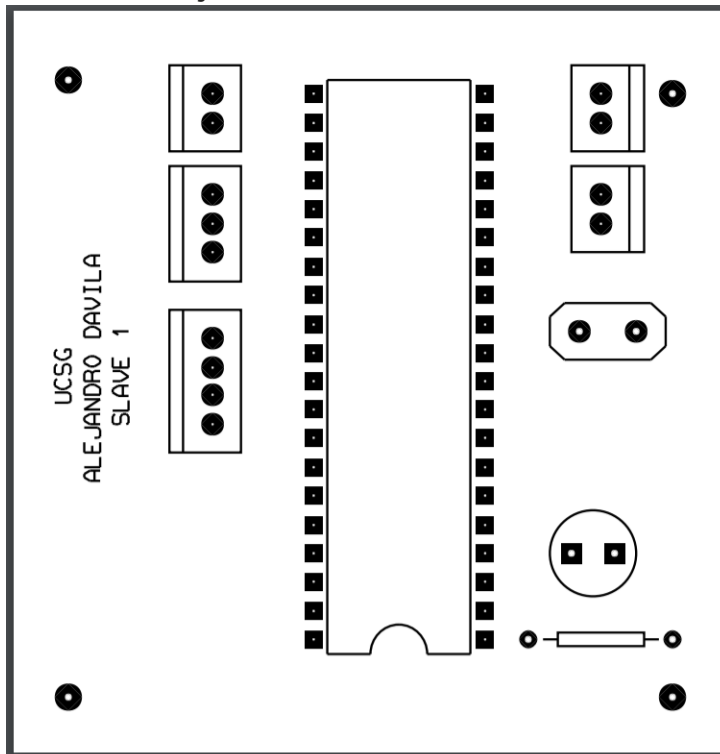


Figura3.18: Tarjeta Slave1 PCB
Elaborado por: Autor

El diseño PCB de la tarjeta electrónica Slave 1 es más simple que la máster, porque lo único que deben ir conectados son los interruptores y un sensor infrarrojo, además como es lógico, el botón de encendido y las conexiones necesarias para poder comunicarse con las otras dos tarjetas, sin tener ningún problema de comunicación.

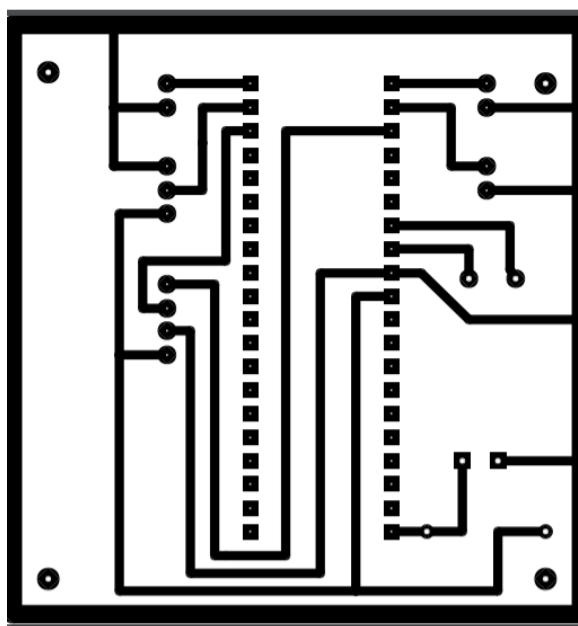


Figura3.19: Tarjeta Slave1 PCB Bottom
Elaborado por: Autor

3.6. Diseño PCB de la tarjeta slave2

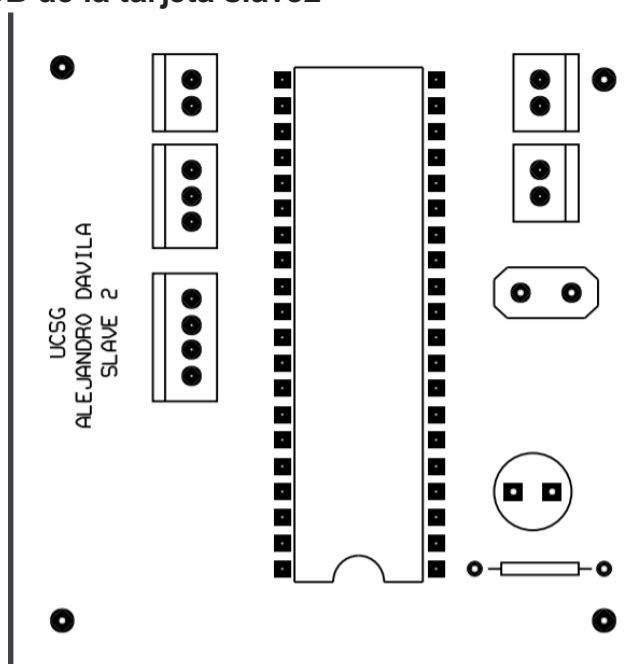


Figura3.20: Tarjeta Slave2 PCB

Elaborado por: Autor

Este diseño al igual que el anterior esclavo es muy similar ya que es el final de todo el sistema. Lo interesante de esta placa es que cómo se conectará a un relé, el mismo que a su vez estará conectado a las salidas, es decir, las luces de la vivienda al igual que las otras tarjetas tienen el espacio suficiente para conectar los elementos necesarios para que la placa funcione correctamente.

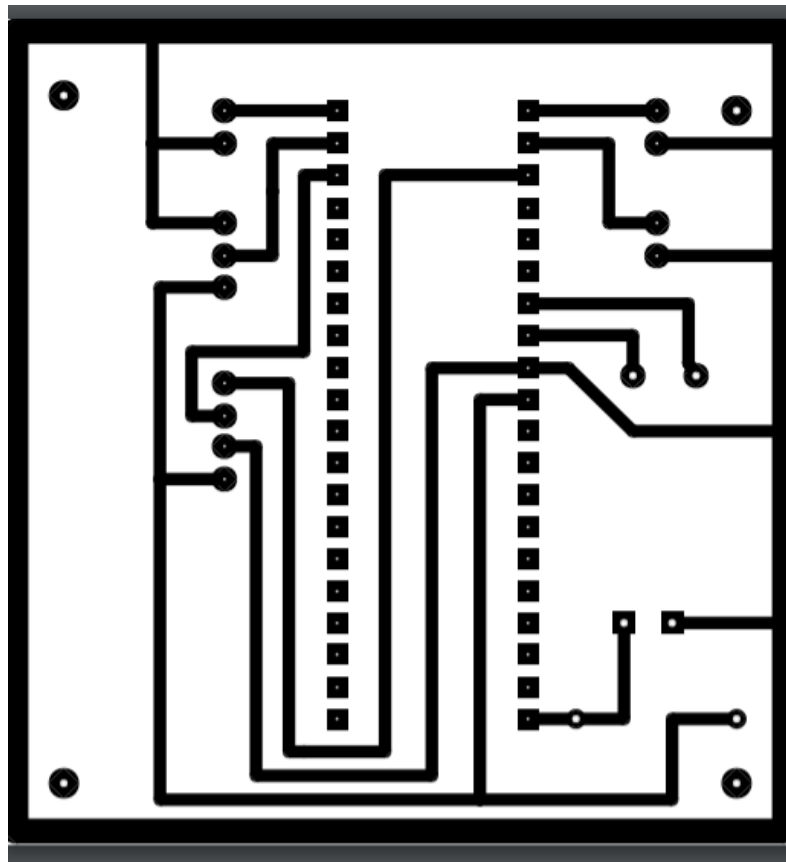


Figura3.21: Tarjeta Slave2 PCB Buttom
Elaborado por: Autor

3.7. Conexiones al Relé de estado solido

Finalmente, en la última placa se instalará a un relé, en cuyo final se conectarán todas las salidas, que en este caso son las luces colocadas en la vivienda; este relé será de gran ayuda porque en él existen leds que notifican cuál de las señales están activas y cuáles no, de esa manera se conoce si el sistema está trabajando correctamente, y por ende se detecta si se presentan daños del sistema de control o si es el foco que ya perdió su vida útil.



Figura3.22: Relé de Estado solido
Elaborado por: Autor

Capítulo 4: Conclusiones y Recomendaciones

4.1. Conclusiones:

A través del presente trabajo de titulación se logró conocer todas las bondades y beneficios que tiene este tipo de tecnología y el protocolo de comunicación que le brindan al usuario; además se logró observar el funcionamiento permitiendo comprobar la confiabilidad de la comunicación con todos los ambientes de la vivienda, al controlar las luces sin ninguna novedad y siguiendo los parámetros establecidos.

En este estudio al aplicarse el protocolo de comunicación KNX con tres tarjetas electrónicas (una tarjeta máster y dos tarjetas denominadas esclavas), permitió comprobar que este protocolo es muy versátil y que puede ser utilizado en cualquier dispositivo de control electrónico.

Finalmente se pudo observar el funcionamiento práctico de las luces led, ubicadas en la implementación y su velocidad de prendido y apagado no provocaron ningún delay notable o que moleste al usuario de la vivienda.

4.2. Recomendaciones:

Antes de instalar este tipo de sistema, se recomienda realizar una inspección de la vivienda y procurar que la instalación sea a cargo de profesionales, debido a que es un equipo sensible, por lo que su manejo y programación debe ser meticulosa.

Para el presente trabajo se manifestó el uso de switches o pulsadores para encender luces, aunque lo más recomendable es utilizar sensores que ayudan con el ahorro de energía.

Para tener un mejor control y que el sistema sea más amigable con el habitante de la vivienda, se podría agregar un HMI, que es un interfaz hombre máquina, que permite conocer por completo el funcionamiento de las tarjetas y cuál de las luminarias se encuentra activada.

Para una instalación de este sistema, se recomienda previamente revisar todo el cableado confirmando su óptimo estado y de esa manera no se generarán problemas de contacto ni de comunicación.

Anexos

Anexo 1
Maqueta para implementación



Fuente: Autor

Anexo 2

Bibliotecas KNX ComObject

```
// This file is part of Arduino Knx Bus Device library.// The Arduino Knx Bus Device library allows to turn Arduino into "self-made" KNX bus device.// Copyright (C) 2014 2015 2016
Franck MARINI (fm@liwan.fr)// The Arduino Knx Bus Device library is free software: you can redistribute it and/or modify// it under the terms of the GNU General Public License as
published by// the Free Software Foundation, either version 3 of the License, or// (at your option) any later version.// This program is distributed in the hope that it will be
useful,// but WITHOUT ANY WARRANTY; without even the implied warranty of// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the// GNU General Public License for more
details.// You should have received a copy of the GNU General Public License// along with this program. If not, see <http://www.gnu.org/licenses/>.// File : KnxComObject.h// Author
: Franck Marini// Description : Handling of the KNX Communication Objects// Module dependencies : KnxTelegram#ifndef KNXCOMOBJECT_H#define KNXCOMOBJECT_H#include "KnxTelegram.h"#include
"KnxDPT.h"// !!!!!!!!!!!!!!!!!!!!! FLAG OPTIONS !!!!!!!!!!!!!!!!!!!!!// By default, all the objects have NORMAL priority, other priorities are not supported// turn KNX_COM_OBJ_SUPPORT_ALL_PRIORITIES
flag on to allow support of all the priorities// #define KNX_COM_OBJ_SUPPORT_ALL_PRIORITIES// Definition of com obj indicator values// See "knx.org" for com obj indicators specification//
INDICATOR field : B7 B6 B5 B4 B3 B2 B1 B0//                xx xx C R W T U I #define KNX_COM_OBJ_C_INDICATOR 0x20 // Communication (C)#define
KNX_COM_OBJ_R_INDICATOR 0x10 // Read (R)#define KNX_COM_OBJ_W_INDICATOR 0x08 // Write (W)#define KNX_COM_OBJ_T_INDICATOR 0x04 // Transmit (T)#define KNX_COM_OBJ_U_INDICATOR 0x02
// Update (U)#define KNX_COM_OBJ_I_INDICATOR 0x01 // Init Read (I)// Definition of predefined com obj profiles// Sensor profile : COM_OBJ_SENSOR#define COM_OBJ_SENSOR
KNX_COM_OBJ_C_R_T_INDICATOR#define KNX_COM_OBJ_C_R_T_INDICATOR 0x34 // ( Communication | Read | Transmit )// Logic input profile : COM_OBJ_LOGIC_IN#define COM_OBJ_LOGIC_IN
KNX_COM_OBJ_C_W_U_INDICATOR#define KNX_COM_OBJ_C_W_U_INDICATOR 0x2A // ( Communication | Write | Update )// Logic input to be initialized at bus power-up profile : COM_OBJ_LOGIC_IN_INIT
#define COM_OBJ_LOGIC_IN_INIT KNX_COM_OBJ_C_W_U_I_INDICATOR#define KNX_COM_OBJ_C_W_U_I_INDICATOR 0x2B // ( Communication | Write | Update | Init)#define KNX_COM_OBJECT_OK 0#define
KNX_COM_OBJECT_ERROR 255class KnxComObject { const word _addr; // Group Address value const byte _dptId; // Datapoint type const byte _indicator; // C/R/W/T/U/I indicators
// Com object data length is calculated in the same way as telegram payload length // (See "knx.org" telegram specification for more details) const byte _length;#ifdef
KNX_COM_OBJ_SUPPORT_ALL_PRIORITIES const e_KnxPriority _prio; // priority#endif // _validity is used for "InitRead" typed com objs : // it remains "false" till the object value
is updated // NB : the objects not typed "InitRead" get "true" validity value boolean _validity; union { // field used in case of short value (1 byte max width, i.e.
length <= 2) struct{ byte _value; byte _notUsed; }; // field used in case of long value (2 bytes width or more, i.e.
length > 2) // The data space is allocated dynamically by the constructor byte *_longValue; }; public: // Constructor :#ifdef
KNX_COM_OBJ_SUPPORT_ALL_PRIORITIES KnxComObject(word addr, e_KnxDPT_ID dptId, e_KnxPriority prio, byte indicator );#else KnxComObject(word addr, e_KnxDPT_ID dptId, byte
indicator );#endif // Destructor ~KnxComObject(); // INLINED functions (see definitions later in this file) word GetAddr(void) const; byte GetDptId(void) const;
e_KnxPriority GetPriority(void) const; byte GetIndicator(void) const; boolean GetValidity(void) const; byte GetLength(void) const; // Return the com obj value (short value
case only) byte GetValue(void) const; // Update the com obj value (short value case only) // Return ERROR if the com obj is long value (invalid use case), else return OK byte
UpdateValue(byte newVal); // Toggle the binary value (for com objs with "B1" format) // NB : the function does not change the validity. void ToggleValue(void); //
functions NOT INLINED : // Get the com obj value (short and long value cases) void GetValue(byte dest[]) const; // Update the com obj value (short and long value cases) void
UpdateValue(const byte ori[]); // Update the com obj value with a telegram payload content // Return ERROR if the telegram payload length differs from com obj one, else return OK byte
UpdateValue(const KnxTelegram& ori); // Copy the com obj attributes (addr, prio, length) into a telegram object void CopyAttributes(KnxTelegram& dest) const; // Copy the com obj
value into a telegram object void CopyValue(KnxTelegram& dest) const; // DEBUG function void Info(String&) const;};// ----- Definition of the INLINED functions
-----#ifndef KNX_COM_OBJ_SUPPORT_ALL_PRIORITIES inline word KnxComObject::GetAddr(void) const { return _addr; }inline byte KnxComObject::GetDptId(void) const { return _dptId; }#endif
inline e_KnxPriority KnxComObject::GetPriority(void) const { return _prio; }#elseinline e_KnxPriority KnxComObject::GetPriority(void) const { return KNX_PRIORITY_NORMAL_VALUE; }#endif
inline byte KnxComObject::GetIndicator(void) const { return _indicator; }inline boolean KnxComObject::GetValidity(void) const { return _validity; }inline byte KnxComObject::GetLength(void)
const { return _length; }inline byte KnxComObject::GetValue(void) const { return _value; } inline byte KnxComObject::UpdateValue(byte newValue) { if (_length > 2) return
KNX_COM_OBJECT_ERROR; _value = newValue; _validity = true; return KNX_COM_OBJECT_OK; }inline void KnxComObject::ToggleValue(void) { _value = !_value; }#endif // KNXCOMOBJECT_H
```

Fuente: Knx Software

Anexo 3 Bibliotecas KNX Device

```
|
#ifndef KNXDEVICE_H
#define KNXDEVICE_H

#include "KnxTelegram.h"
#include "KnxComObject.h"
#include "ActionRingBuffer.h"
#include "KnxTpUart.h"

// !!!!!!!!!!!!!!!!!!!!! FLAG OPTIONS !!!!!!!!!!!!!!!!!!!!!
// DEBUG :
// #define KNXDEVICE_DEBUG_INFO // Uncomment to activate info traces

// Values returned by the KnxDevice member functions :
enum e_KnxDeviceStatus {
    KNX_DEVICE_OK = 0,
    KNX_DEVICE_NOT_IMPLEMENTED = 254,
    KNX_DEVICE_ERROR = 255
};

// Macro functions for conversion of physical and 2/3 level group addresses
inline word P_ADDR(byte area, byte line, byte busdevice)
{ return (word) ( ((area&0xF)<<12) + ((line&0xF)<<8) + busdevice ); }

inline word G_ADDR(byte maingrp, byte midgrp, byte subgrp)
{ return (word) ( ((maingrp&0x1F)<<11) + ((midgrp&0x7)<<8) + subgrp ); }

inline word G_ADDR(byte maingrp, byte subgrp)
{ return (word) ( ((maingrp&0x1F)<<11) + subgrp ); }

#define ACTIONS_QUEUE_SIZE 16

// KnxDevice internal state
enum e_KnxDeviceState {
    INIT,
    IDLE,
    TX_ONGOING,
};

// Action types
enum e_KnxDeviceTxActionType {
    EIB_READ_REQUEST,
    EIB_WRITE_REQUEST,
    EIB_RESPONSE_REQUEST
};
```

Fuente: Knx Software

Bibliografía

Aguirre Quiroz, M. F. (2011).

Aguirre Quiroz, S. S., & Mogollón Flores, E. D. (2011). *Diseño e implementación del sistema inmótico para el control de iluminación en el aeropuerto de Latacunga basado en la tecnología Lonworks* (Bachelor's thesis, SANGOLQUÍ/ESPE/2011).

Carlos Daniel Ramirez, R. S. (2 de Septiembre de 2011). *Uniminuto*. Obtenido de Repository Uniminuto.edu:
https://repository.uniminuto.edu/bitstream/handle/10656/1785/TE_RamirezCarlosDaniel_2011.pdf?sequence=1&isAllowed=y

Cabezas García, W., Neira, J., & Amadeo, F. (2016). Diseño de un control domótico basado en una plataforma Open Source para viviendas.

Cabot Feliu, C. (2017). Desarrollo e implementación de un sistema basado en equipos KNX.

CARRICONDO MONTER, A. I. T. O. R. (2017). *Desarrollo de un sistema de monitorización domiciliaria basado en la plataforma NodeMCU V3* (Doctoral dissertation).

Cifuentes Castillo, E. E., Rodríguez, R., Fredy, W., & Romero Zipa, S. M. (2018). *Estudio de prefactibilidad para implementar sistemas domóticos en las casa del sector de La Española en la ciudad de Bogotá para potenciar el bienestar humano y el medio ambiente* (Doctoral dissertation, Corporación Universitaria Minuto de Dios).

Cunalata, S., & Fabricio, D. (2016). *Diseño e implementación de un sistema de domótica basado en la tecnología smart bus KNX para el control de iluminación, audio y seguridad, mediante un enlace web Apps* (Bachelor's thesis, Quito, 2016.).

Cunalata, S., & Fabricio, D. (2016). *Diseño e implementación de un sistema de domótica basado en la tecnología smart bus KNX para el control de iluminación, audio y seguridad, mediante un enlace web Apps* (Bachelor's thesis, Quito, 2016.).

E-Marmolejo., D. R. (12 de noviembre de 2017). *hetpro*. Obtenido de hetpro:
<https://hetpro-store.com/TUTORIALES/microcontrolador/>

Gordon Sánchez, B. R., Romero, T., & Gustavo, E. (2016). *Diseño e implementación de un módulo didáctico de domótica por medio de láminas con la tarjeta raspberry PI y el programa QT en la universidad politécnica salesiana sede Guayaquil*(Bachelor's thesis).

Hernandez, A. (30 de Noviembre de 2014). *Taller Electronica*. Obtenido de Taller Electronica : <https://tallerelectronica.com/2014/11/30/microcontroladores/>

Jara Maldonado, P. A. (2015). *Estudio y diseño de un sistema inmótico para seguridad, comunicación y confort, utilizando el protocolo KNX para el edificio Torre Piamonte ubicado en el sector de Totoracocha de la ciudad de Cuenca* (Bachelor's thesis).

Jhon, D. (26 de marzo de 2018). *Diferenciaentre*. Obtenido de Diferenciaentre:
<http://www.diferenciaentre.net/diferencia-entre-microprocesador-y-microcontrolador/>

- KNX. (2018). *knx.org*. Obtenido de KNX: <https://www2.knx.org/co/knx/associacion/que-es-knx/index.php>
- Monter, C. (septiembre de 2017). *riunet.upv.es*. Obtenido de riunet.upv.es: <https://riunet.upv.es/bitstream/handle/10251/92938/CARRICONDO%20%20Desarrollo%20de%20un%20sistema%20de%20monitorizaci%C3%B3n%20domiciliaria%20basado%20en%20la%20plataforma%20Node....pdf?sequence=1&isAllowed=y>
- López, M. E. A., & Rios, L. J. (2018). Gestión del control de acceso con tecnología open source en proyectos de domótica. In *XX Workshop de Investigadores en Ciencias de la Computación (WICC 2018, Universidad Nacional del Nordeste)*.
- Lozano, F., & Arturo, A. (2016). *Diseño de un sistema domótico para el acceso a las aulas de la Facultad de Ingeniería Industrial de la Universidad de Guayaquil* (Bachelor's thesis, Universidad de Guayaquil. Facultad de Ingeniería Industrial. Carrera de Licenciatura en Sistemas de Información).
- Muñoz, C. A. O., & Guryev, I. (2017). DESARROLLO DEL SISTEMA DE CONTROL AUTÓNOMO DE LA ILUMINACIÓN PARA DOMÓTICA. *JÓVENES EN LA CIENCIA*, 2(1), 1033-1036.
- Muñoz, A. (29 de Octubre de 2016). *computerhoy*. Obtenido de computerhoy: <https://computerhoy.com/noticias/software/que-es-firmware-53182>
- Pedro, J. (2015). *universidad saleciana*. Obtenido de ups.
- Pedraza Cordones, Á. (2017). GITT." Diseño de una Arquitectura Software Basada en el Paradigma del Internet de las Cosas: Implementación en un Sistema Domótico".
- Peirotén, M. (2013). Sistema domótico para una casa inteligente. *Madrid: Escuela Técnica Superior de Ingeniería (ICAI) Ingeniero Industrial*.
- Pomaquíza, L., & Javier, E. (2016). *Diseño e implementación de un sistema domótico con acceso remoto utilizando una pasarela residencial* (Bachelor's thesis, Escuela Superior Politécnica de Chimborazo).
- Ramos, V. M. J., Julián, F. G. C., Baltazar, R. T. C., & Pérez, A. F. (2016). DESARROLLO MODULAR DE UN MODELO GENÉRICO PARA LA DOMÓTICA MEDIANTE COMUNICACIÓN INALÁMBRICA BASADA EN EL PROTOCOLO ZIGBEE. *Pistas Educativas*, 38(122).
- RAMIREZ, C. D. (s.f.).
- Valencia, C. D. (2016). *core*. Obtenido de core.ac.uk: <https://core.ac.uk/download/pdf/71399297.pdf>
- Rodríguez Diego, M. (2017). Gestión domótica de una casa unifamiliar basada en Arduino.
- Ruiz Garcia, M., & Calle Sánchez, S. (2016). *Sistema de gestión domótico para una vivienda* (Bachelor's thesis, Universitat Politècnica de Catalunya).
- Sánchez, C., Mesa, A., Manrique, C., Calderón, H., Cobo, L., Dorado, R., & Mejía, C. (2015). Diseño e implementación de un prototipo de vivienda domótica basado en las plataformas arduino y android. *Revista Ontare*, 2(2), 115-132.

- Valencia, C. D. L. (2016). *Estado de la domótica en Pereira*(Doctoral dissertation, Universidad Tecnológica de Pereira. Facultad de Ingenierías Eléctrica, Electrónica, Física y Ciencias de la Computación. Ingeniería Eléctrica.).
- Valer, F. S., & Rodríguez, A. C. (2017). Domótica mediante la transmisión de datos digitales por la red eléctrica usando protocolo X10. *Perfiles de Ingeniería, 13*(13), 171-182.
- Varela, C. F., Medina, J. A. B., & Acevedo, A. V. R. (2008). Diseño de un modulo para el control de iluminación de un hardware domótico a través de un teléfono celular en un ambiente distribuido. *Revista de Tecnología, 7*(1).
- Vega, N., Parra, P., & Martillo, D. (2017, October). Didactic equipment developed with embedded systems for the learning of engineering. In *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)* (pp. 1-6). IEEE.
- Vega, N., Parra, P., Córdova, L., Andramuño, J., & Navarrete, V. (2018, October). Adaptive Control Implemented with Embedded Systems. In *2018 IEEE International Conference on Automation/XXIII Congress of the Chilean Association of Automatic Control (ICA-ACCA)* (pp. 1-6). IEEE.
- VERDEGUER, M. B. (2016). *Gestión de iluminación mediante protocolo de gestión KNX en almacén de alimentación* (Doctoral dissertation).
- Vieira, A., Blanco, X., & Quijadas, D. (2018). Sistema domótico para control de temperatura e iluminación de un apartamento para lesionados medulares (parapléjicos). *Tekhné, 21*(5).
- Wilber, C. G. (2016). *riunet.upv.es*. Obtenido de riunet.upv.es:
 file:///C:/Users/USUARIO_01/Downloads/Wilber_Tesis_T%C3%ADtuloprofesional_2016.pdf



DECLARACIÓN Y AUTORIZACIÓN

Yo, **DÁVILA CHEVALIER ALEJANDRO ANDRÉS** con C.C: # 0920164514 autor del Trabajo de Titulación: **Aplicación del protocolo de comunicación KNX en el control de luces de una vivienda unifamiliar**, previo a la obtención del título de **INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 20 de marzo de 2019

f. _____

Nombre: DÁVILA CHEVALIER ALEJANDRO ANDRÉS

C.C: # 0920164514



Presidencia
de la República
del Ecuador



Plan Nacional
de Ciencia, Tecnología,
Innovación y Saberes



SENESCYT
Secretaría Nacional de Educación Superior,
Ciencia, Tecnología e Innovación

| REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA | | |
|--|---|--|
| FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN | | |
| TÍTULO Y SUBTÍTULO: | Aplicación del protocolo de comunicación KNX en el control de luces de una vivienda unifamiliar. | |
| AUTOR(ES) | Dávila Chevalier, Alejandro Andrés | |
| REVISOR(ES)/TUTOR(ES) | ING. CÓRDOVA RIVADENEIRA, LUIS MSC. | |
| INSTITUCIÓN: | Universidad Católica de Santiago de Guayaquil | |
| FACULTAD: | Facultad de Educación Técnica para el Desarrollo | |
| CARRERA: | Ingeniería Electrónica en Control y Automatismo | |
| TÍTULO OBTENIDO: | Ingeniero Electrónico en Control y Automatismo | |
| FECHA DE PUBLICACIÓN: | 20 de marzo del 2019 | No. DE PÁGINAS 75 |
| ÁREAS TEMÁTICAS: | Domotica, Microcontroladores | |
| PALABRAS CLAVES/ KEYWORDS: | DOMOTICA, KNX, ARDUINO, RASPBERRY, SWITCHES, PROTOCOLO DE COMUNICACIÓN | |
| RESUMEN/ABSTRACT: | <p>Al aparecer los dispositivos inteligentes como los ordenadores y los teléfonos celulares se fueron ideando una meta la cual era la automatización de un hogar, a lo cual se refiere a encendido y apagado de dispositivos de instalaciones eléctricas y electrónicas de forma centralizada o remota. El presente trabajo de titulación tiene como finalidad presentar una aplicación demótica mediante el protocolo KNX, el cual explica su elevada relevancia, especialmente en la era tecnológica en la que nos encontramos. Para la investigación se eligió el protocolo de comunicación KNX debido a la practicidad de comunicación, por ser compatible con arduinos, raspberries y pics, además de y por la confiabilidad de sus elementos como los switches de luz. Los resultados de este trabajo de investigación corroboran las ventajas de la inversión y la confiabilidad del tipo de sistema en las viviendas de actualidad.</p> | |
| ADJUNTO PDF: | <input checked="" type="checkbox"/> si | <input type="checkbox"/> no |
| CONTACTO CON AUTOR/ES: | Teléfono: 593-988716541 | E-mail: alejo.davila19@gmail.com |
| CONTACTO CON LA INSTITUCIÓN: | Nombre: Mendoza Merchán, Eduardo Vicente | |
| COORDINADOR DEL PROCESO DE UTE | Teléfono: +593-9-85086815 | |
| | E-mail: eduardo.mendoza@cu.ucsg.edu.ec | |
| SECCIÓN PARA USO DE BIBLIOTECA | | |
| No. DE REGISTRO (en base a datos): | | |
| No. DE CLASIFICACIÓN: | | |
| DIRECCIÓN URL (tesis en la web): | | |