



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería Mecánica Eléctrica

**DISEÑO DE UN SISTEMA DE AUTOMATIZACIÓN DE UNA CASA UTILIZANDO
MICROCONTROLADOR PIC16F877A EN COMUNICACIÓN BLUETHOOTH CON UN
TELÉFONO CELULAR CON PLATAFORMA ANDROID**

Juan José Zetino González

Asesorado por la Inga. Ingrid Salomé Rodríguez de Loukota

Guatemala, enero de 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO DE UN SISTEMA DE AUTOMATIZACIÓN DE UNA CASA UTILIZANDO
MICROCONTROLADOR PIC16F877A EN COMUNICACIÓN BLUETHOOH CON UN
TELÉFONO CELULAR CON PLATAFORMA ANDROID**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

JUAN JOSÉ ZETINO GONZÁLEZ

ASESORADO POR LA INGA. INGRID SALOMÉ RODRÍGUEZ DE LOUKOTA

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN ELECTRÓNICA

GUATEMALA, ENERO DE 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

| | |
|------------|--|
| DECANO | Ing. Pedro Antonio Aguilar Polanco |
| VOCAL I | Ing. Angel Roberto Sic García |
| VOCAL II | Ing. Pablo Christian de León Rodríguez |
| VOCAL III | Inga. Elvia Miriam Ruballos Samayoa |
| VOCAL IV | Br. Raúl Eduardo Ticún Córdova |
| VOCAL V | Br. Henry Fernando Duarte García |
| SECRETARIA | Inga. Lesbia Magalí Herrera López |

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

| | |
|-------------|-------------------------------------|
| DECANO | Ing. Murphy Olympo Paiz Recinos |
| EXAMINADOR | Ing. Julio Rolando Barrios Archila |
| EXAMINADOR | Ing. Julio César Solares Peñate |
| EXAMINADORA | Inga. María Magdalena Puente Romero |
| SECRETARIO | Ing. Hugo Humberto Rivera Pérez |

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

DISEÑO DE UN SISTEMA DE AUTOMATIZACIÓN DE UNA CASA UTILIZANDO MICROCONTROLADOR PIC16F877A EN COMUNICACIÓN BLUETHOOOTH CON UN TELÉFONO CELULAR CON PLATAFORMA ANDROID

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería Mecánica Eléctrica, con fecha 4 de febrero de 2013.



Juan José Zetino Gonzalez

Guatemala 11 de febrero de 2015

Ingeniero
Carlos Eduardo Guzmán Salazar
Coordinador del Área de Electrónica
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería, USAC.

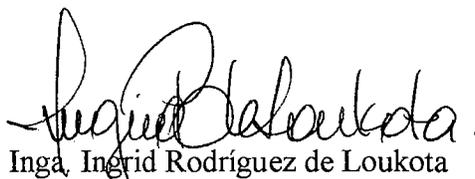
Estimado Ingeniero Guzmán.

Me permito dar aprobación al trabajo de graduación titulado: **Diseño de un sistema de automatización de una casa utilizando microcontrolador PIC16F877A en comunicación Bluetooth con un teléfono celular con plataforma Android**, del señor **Juan José Zetino González**, por considerar que cumple con los requisitos establecidos.

Por tanto, el autor de este trabajo de graduación y, yo, como su asesora, nos hacemos responsables por el contenido y conclusiones del mismo.

Sin otro particular, me es grato saludarle.

Atentamente,



Inga, Ingrid Rodríguez de Loukota
Colegiada 5,356
Asesora

Ingrid Rodríguez de Loukota
Ingeniera en Electrónica
colegiado 5356



Ref. EIME 50. 2015

Guatemala, 13 de JULIO 2015.

Señor Director
Ing. Guillermo Antonio Puente Romero
Escuela de Ingeniería Mecánica Eléctrica
Facultad de Ingeniería, USAC.

Señor Director:

Me permito dar aprobación al trabajo de Graduación titulado:
**DISEÑO DE UN SISTEMA DE AUTOMATIZACIÓN DE UNA
CASA UTILIZANDO MICROCONTROLADOR PIC16F877A EN
COMUNICACIÓN BLUETHOOTH CON UN TELÉFONO
CELULAR CON PLATAFORMA ANDROID,** del estudiante
Juan José Zetino González que cumple con los requisitos
establecidos para tal fin.

Sin otro particular, aprovecho la oportunidad para saludarle.

Atentamente,
ID Y ENSEÑADA TODOS

Ing. Carlos Eduardo Guzmán Salazar
Coordinador Área Electrónica



SFO



REF. EIME 50. 2015.

El Director de la Escuela de Ingeniería Mecánica Eléctrica, después de conocer el dictamen del Asesor, con el Visto Bueno del Coordinador de Área, al trabajo de Graduación del estudiante; JUAN JOSÉ ZETINO GONZÁLEZ titulado: DISEÑO DE UN SISTEMA DE AUTOMATIZACIÓN DE UNA CASA UTILIZANDO MICROCONTROLADOR PIC16F877A EN COMUNICACIÓN BLUETHOOTH CON UN TELÉFONO CELULAR CON PLATAFORMA ANDROID, procede a la autorización del mismo.

Ing. Guillermo Antonio Puente Romero

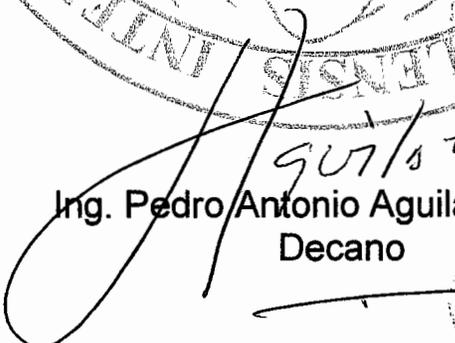


GUATEMALA, 31 DE JULIO 2,015.



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Mecánica Eléctrica al trabajo de graduación titulado **DISEÑO DE UN SISTEMA DE AUTOMATIZACIÓN DE UNA CASA UTILIZANDO MICROCONTROLADOR PIC-16F877A EN COMUNICACIÓN BLUETHOOTH CON UN TELÉFONO CELULAR CON PLATAFORMA ANDROID**, presentado por el estudiante universitario **Juan José Zetino González**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.


Ing. Pedro Antonio Aguilar Polanco
Decano



Guatemala, enero 2016

ACTO QUE DEDICO A:

- Dios** Por darme la vida y fuerza necesaria para culminar mi carrera y llenarme de bendiciones.
- Mis padres** Juan José Zetino Osorio (q. e. p. d.) y Ana Maclovia González de Zetino, por ser los pilares de mi vida y enseñarme el valor del trabajo. Por brindarme en todo momento su amor y apoyo. Los amo padres.
- Mis hermanas** Rosita y Margarita Zetino, por estar siempre a mi lado y por apoyarme, cuidarme en todo momento y ser la fuerza que me ayuda a seguir.
- Mis sobrinas y sobrinos** Silvia Castillo, Guadalupe Castillo, Otto Pérez Castillo, Sebastián Ramírez, María Fátima Ramírez, Alejandro Chávez, Josué Castillo y Daniel Chávez, por el amor que siempre me muestran y espero ser un buen ejemplo en su vida.
- Mis tías y primos** Por las muestras de cariño que me han dado en todo momento. Muy en especial a Rubén Solares, por ser como un hermano para mí.

Mis amigos

Edgar Rodríguez, Renato Rodríguez, Mynor Ramírez, Jaime Maldonado, Jorge Paredes, Guillermo Soto, Geovani Velásquez, Iván Cansinos, Moisés Recinos, Edwin Marroquín, Alvin Reyes, Claudia Bautista, Lorena Ramírez, Gladis Ramírez, Nicolle Ramírez y Judith Ramírez, por todas las veces que han estado cuándo los necesitaba y brindarme su ayuda.

Verónica Ramírez

Gracias por ser esa persona que siempre me apoyó y ayudó para llegar al final de mi carrera, te quiero mucho.

AGRADECIMIENTOS A:

| | |
|---|--|
| Universidad de San Carlos de Guatemala | Casa de estudios que me abrió las puertas para formarme como un profesional. |
| Facultad de Ingeniería | Por brindarme los conocimientos necesarios para que pueda ejercer una carrera profesional. |
| Mis amigos de la Facultad | Por todas las batallas que luchamos juntos y las buenas experiencias que vivimos. |
| Las empresas | JL Betancourt e Intecap, por darme una fuente de trabajo y la oportunidad de estudiar. |
| Mi asesora de trabajo de graduación | Inga. Ingrid Salomé Rodríguez de Loukota, por el apoyo brindado a mi persona y el tiempo que le dedicó a mi trabajo de graduación. |

ÍNDICE GENERAL

| | |
|---|-------|
| ÍNDICE DE ILUSTRACIONES | IX |
| LISTA DE SÍMBOLOS | XIII |
| GLOSARIO | XV |
| RESUMEN | XIX |
| OBJETIVOS..... | XXI |
| INTRODUCCIÓN | XXIII |
| | |
| 1. AUTOMATIZACIÓN DE LAS FUNCIONES DE UNA CASA | 1 |
| 1.1. La domótica | 2 |
| 1.1.1. Elementos de un sistema domótico | 3 |
| 1.1.2. Arquitectura de los sistemas domóticos | 5 |
| 1.1.2.1. Arquitectura centralizada | 5 |
| 1.1.2.2. Arquitectura descentralizada | 6 |
| 1.1.2.3. Arquitectura distribuida | 6 |
| 1.1.2.4. Arquitectura híbrida o mixta | 6 |
| 1.1.3. Medios de transmisión | 7 |
| 1.1.4. Protocolos de comunicación en la domótica..... | 7 |
| 1.1.4.1. X-10 | 8 |
| 1.1.4.2. EIB (European Installation Bus)..... | 10 |
| 1.1.4.3. <i>Konnex</i> | 11 |
| 1.1.4.4. <i>Lonworks</i> | 12 |
| 1.1.4.5. <i>JINI</i> | 13 |
| 1.1.4.6. TCP/IP | 13 |
| 1.1.5. Ventajas de la domótica | 14 |
| 1.1.5.1. <i>Confort</i> | 14 |

| | | |
|------------|---|----|
| 1.1.5.2. | Seguridad | 15 |
| 1.1.5.3. | Ahorro energético | 17 |
| 1.1.6. | Desventajas de la domótica | 18 |
| 1.1.7. | Identificación de las funciones para automatizar una casa | 20 |
| 1.1.7.1. | Control de iluminación | 22 |
| 1.1.7.1.1. | Tipos de lámparas..... | 22 |
| 1.1.7.2. | Apertura de puertas..... | 24 |
| 1.1.7.2.1. | Tipos de cerraduras | 25 |
| 1.1.7.3. | Portón eléctrico | 27 |
| 1.1.7.4. | Apertura de ventanas | 29 |
| 1.1.7.4.1. | Apertura de persianas de ventanas..... | 29 |
| 1.1.7.5. | Regado de jardín..... | 30 |
| 1.1.7.6. | Control perimetral de la casa..... | 31 |
| 1.1.7.6.1. | Sensores de movimiento..... | 32 |
| 1.1.7.6.2. | Sensores magnéticos.... | 33 |
| 2. | ELEMENTOS DEL CONTROL | 35 |
| 2.1. | El microcontrolador PIC16F877A..... | 36 |
| 2.1.1. | Arquitectura del microcontrolador..... | 36 |
| 2.1.1.1. | Arquitectura interna de microcontrolador..... | 36 |
| 2.1.1.2. | Arquitectura externa del microcontrolador..... | 38 |
| 2.1.2. | Memoria interna | 40 |
| 2.1.2.1. | Memoria de datos..... | 40 |
| 2.1.2.1.1. | Memoria RAM | 40 |

| | | | | |
|---------|------------|---|--|----|
| | 2.1.2.1.2. | Memoria EEPROM | 42 | |
| | 2.1.2.2. | Memoria de programa | 42 | |
| 2.1.3. | | Características básicas del microcontrolador PIC16F877A | 43 | |
| 2.1.4. | | Alimentación del microcontrolador | 44 | |
| 2.1.5. | | El oscilador externo | 44 | |
| 2.1.6. | | <i>Reset</i> | 46 | |
| | 2.1.6.1. | Encendido del reset | 46 | |
| 2.1.7. | | Temporizadores | 47 | |
| 2.1.8. | | Interrupciones | 48 | |
| 2.1.9. | | Puerto serial | 50 | |
| 2.1.10. | | Programación del microcontrolador | 52 | |
| | 2.1.10.1. | Lenguaje de programación <i>BASIC</i> | 54 | |
| 2.2. | | Sensores | 55 | |
| | 2.2.1. | Clasificación de sensores según su tipo de señal de salida | 57 | |
| | 2.2.2. | Sensores pasivos y activos | 58 | |
| | 2.2.3. | Características generales de los sensores | 58 | |
| | | 2.2.3.1. Características estáticas | 59 | |
| | | 2.2.3.2. Características dinámicas | 60 | |
| | | | 2.2.3.2.1. Velocidad de respuesta | 61 |
| | | | 2.2.3.2.2. Constante de tiempo | 61 |
| | | | 2.2.3.2.3. Respuesta frecuencial .. | 62 |
| | | | 2.2.3.2.4. Estabilidad y derivas | 62 |
| 2.3. | | Actuadores | 62 | |
| | 2.3.1. | Actuadores eléctricos | 62 | |
| | | 2.3.1.1. Motor eléctrico | 65 | |

| | | | |
|------|------------|--|-----|
| | 2.3.1.1.1. | Motor eléctrico de corriente alterna monofásico..... | 66 |
| | 2.3.1.1.2. | Motor eléctrico de corriente directa | 70 |
| | 2.3.1.1.3. | Electroválvula..... | 72 |
| 2.4. | | Comunicación Bluetooth | 73 |
| | 2.4.1. | Módulos de comunicación Bluetooth..... | 76 |
| 3. | | SISTEMA ANDROID..... | 79 |
| 3.1. | | Historia | 80 |
| 3.2. | | Arquitectura..... | 81 |
| | 3.2.1. | Núcleo Linux..... | 81 |
| | 3.2.2. | Bibliotecas | 82 |
| | 3.2.3. | Marco de trabajo de aplicaciones..... | 84 |
| | 3.2.4. | Aplicaciones | 86 |
| 3.3. | | Versiones del sistema Android..... | 87 |
| 3.4. | | Equipos que utilizan sistema Android..... | 96 |
| | 3.4.1. | <i>Smartphone</i> | 97 |
| | 3.4.2. | <i>Tablet</i> | 99 |
| | 3.4.3. | <i>Netbook</i> | 101 |
| | 3.4.4. | <i>Smart TV</i> | 102 |
| | 3.4.4.1. | Android TV | 103 |
| | 3.4.5. | <i>Smartwatch</i> | 104 |
| 3.5. | | Aplicaciones Android..... | 105 |
| | 3.5.1. | <i>Framework</i> de aplicaciones..... | 106 |
| | 3.5.2. | Java..... | 107 |
| | 3.5.3. | Software para realizar aplicaciones Android | 108 |

| | | |
|------------|---|-----|
| 3.5.3.1. | SDK | 108 |
| 3.5.3.2. | Eclipse | 109 |
| 3.5.3.2.1. | Obtener, instalar y ejecutar eclipse IDE.... | 111 |
| 3.5.3.3. | APP Inventor..... | 112 |
| 3.5.3.3.1. | Panel de diseño..... | 114 |
| 3.5.3.3.2. | Editor de bloques..... | 117 |
| 3.5.3.3.3. | Utilizar el emulador..... | 119 |
| 3.5.3.3.4. | Elementos de programación en App Inventor | 119 |
| 4. | DISEÑO DEL SISTEMA DE AUTOMATIZACIÓN | 125 |
| 4.1. | Elementos que se van a controlar en la casa | 125 |
| 4.1.1. | Iluminación..... | 125 |
| 4.1.2. | Puertas y portón eléctrico | 126 |
| 4.1.3. | Apertura de persianas de las ventanas | 126 |
| 4.1.4. | Regado del jardín | 126 |
| 4.1.5. | Control perimetral de la casa | 126 |
| 4.2. | Diseño del control | 127 |
| 4.2.1. | Conexión de los actuadores | 127 |
| 4.2.1.1. | Lámparas..... | 127 |
| 4.2.1.2. | Cerraduras de puerta..... | 128 |
| 4.2.1.3. | Motor de portón | 130 |
| 4.2.1.4. | Motores en persianas de ventanas | 132 |
| 4.2.1.5. | Electroválvula hidráulica | 134 |
| 4.2.2. | Conexión de sensores | 135 |
| 4.2.2.1. | Sensores magnéticos | 135 |
| 4.2.2.2. | Sensor de movimiento | 137 |

| | | |
|----------|---|-----|
| 4.2.3. | Circuitos de potencia | 139 |
| 4.2.3.1. | Circuitos de iluminación..... | 139 |
| 4.2.3.2. | Circuitos de motor alterno | 140 |
| 4.2.3.3. | Circuito de potencia para cerraduras y electroválvula..... | 141 |
| 4.2.4. | Circuito de comunicación | 142 |
| 4.2.5. | Circuitos de mando | 146 |
| 4.2.5.1. | Programa del microcontrolador | 148 |
| 4.3. | Diseño de la aplicación Android para la interfase hombre máquina (HMI) | 151 |
| 4.3.1. | Programación en lenguaje Java a través del software APP Inventor..... | 151 |
| 4.3.2. | Instalar y configurar el APP Inventor | 151 |
| 4.3.3. | Programar la aplicación con APP Inventor | 152 |
| 4.3.4. | Implementación de aplicación desarrollada en el teléfono celular | 165 |
| 4.4. | Evaluación del funcionamiento del sistema de automatización | 165 |
| 4.4.1. | Iluminación de la casa | 166 |
| 4.4.2. | Apertura de puertas..... | 167 |
| 4.4.3. | Portón eléctrico | 169 |
| 4.4.4. | Apertura de persianas de ventanas..... | 170 |
| 4.4.5. | Regado de jardín..... | 171 |
| 4.4.6. | Seguridad perimetral | 172 |
| 4.5. | Costos del sistema propuesto | 173 |
| 4.5.1. | Costos de los materiales | 173 |
| 4.5.2. | Costos de la fabricación del control e instalación | 175 |
| 4.5.3. | Costo total del proyecto..... | 176 |

| | |
|----------------------|-----|
| CONCLUSIONES | 177 |
| RECOMENDACIONES..... | 179 |
| BIBLIOGRAFÍA..... | 181 |
| ANEXOS..... | 183 |

ÍNDICE DE ILUSTRACIONES

FIGURAS

| | | |
|-----|--|----|
| 1. | Casa inteligente..... | 1 |
| 2. | Elementos de un sistema de domótico..... | 4 |
| 3. | Arquitectura domótica centralizada | 5 |
| 4. | Protocolo X-10..... | 8 |
| 5. | Arquitectura del sistema EIB..... | 10 |
| 6. | Plano de una casa de un nivel | 21 |
| 7. | Lámpara incandescente, inducción electromagnética y led | 23 |
| 8. | Cerradura electromagnética, pestillo, motorizada y eléctrica. | 27 |
| 9. | Portón corredizo | 28 |
| 10. | Persiana motorizada para ventana..... | 30 |
| 11. | Sistema de riego eléctrico | 31 |
| 12. | Sensor de movimiento..... | 33 |
| 13. | Sensores magnéticos para puerta o ventana | 34 |
| 14. | Arquitectura interna del PIC16F877A..... | 38 |
| 15. | Distribución de los puertos del PIC16F877A | 39 |
| 16. | Conexión de un oscilador XT. | 45 |
| 17. | Conexión del botón de <i>reset</i> | 47 |
| 18. | Lógica de funcionamiento de las interrupciones y registro INTCON | 49 |
| 19. | Conexión del microcontrolador 16F877A para hacer una comunicación serial..... | 51 |
| 20. | Ejemplo de un programa en lenguaje BASIC, la compilacion y programación en la memoria de programa..... | 54 |

| | | |
|-----|--|-----|
| 21. | El proceso de la compilación de programa escrito en Basic en código hex..... | 55 |
| 22. | Estructura genérica de un transductor | 57 |
| 23. | Partes de un relé..... | 64 |
| 24. | Motor monofásico con bobinado auxiliar y condensador. | 67 |
| 25. | Esquema de motor monofásico de espira en cortocircuito..... | 68 |
| 26. | Esquema de conexiones del motor universal..... | 69 |
| 27. | Funcionamiento de un motor de corriente continua | 72 |
| 28. | Electroválvula servopilotadas..... | 73 |
| 29. | Comunicación entre elementos vía Bluetooth..... | 76 |
| 30. | Diagrama del modulo RN41 | 78 |
| 31. | Arquitectura del sistema android..... | 87 |
| 32. | Secuencia de programación con APP Inventor..... | 114 |
| 33. | Panel de diseño de App Inventor. | 115 |
| 34. | Editor de bloques de APP Inventor. | 118 |
| 35. | Bloques de sentencias | 120 |
| 36. | Bloque de condiciones | 120 |
| 37. | Bloques de bucles..... | 121 |
| 38. | Bloques de variables..... | 122 |
| 39. | Bloques de eventos..... | 123 |
| 40. | Soportes de instalación para cerraduras electromagnéticas..... | 129 |
| 41. | Motor monofásico de portón corredizo y conexión interna | 132 |
| 42. | Instalacion de persiana motorizada y ajuste de finales de carrera del motor..... | 133 |
| 43. | Electroválvula para riego..... | 135 |
| 44. | Instalación de sensor magnético en puerta..... | 137 |
| 45. | Conexión de un sensor de movimiento | 138 |
| 46. | Control de potencia de luces..... | 139 |
| 47. | Control de potencia de motor AC..... | 141 |

| | | |
|-----|---|-----|
| 48. | Control de potencia para cerradura magnética y electroválvula | 142 |
| 49. | Tarjeta Bluetooth Stick de la empresa Mikroelektronika | 143 |
| 50. | Diagrama esquemático del Bluetooth Stick. | 144 |
| 51. | Circuito de mando principal. | 148 |
| 52. | Página de inicio de APP Inventor | 153 |
| 53. | Página de proyecto de APP Inventor. | 154 |
| 54. | Ventana de diseño de usuario | 155 |
| 55. | Selección de botón y prueba en teléfono | 156 |
| 56. | Pantalla de diseñador. | 157 |
| 57. | Ventana de bloques de APP Inventor. | 158 |
| 58. | Seleccionando bloques | 159 |
| 59. | Diseño de la comunicación Bluetooth. | 160 |
| 60. | Diseño de las distintas pantallas de la aplicación. | 161 |
| 61. | Bloques del control de persianas para ventanas | 162 |
| 62. | Bloques de programación para el control de seguridad | 164 |
| 63. | Conexión con el módulo Bluetooth. | 166 |
| 64. | Iluminación de la casa | 167 |
| 65. | Control de cerraduras magnéticas. | 168 |
| 66. | Apertura y cerrado del portón. | 169 |
| 67. | Control de motor de persiana de ventana | 170 |
| 68. | Bloques de programación para el control de seguridad | 171 |
| 69. | Control de seguridad perimetral de la casa | 173 |

TABLAS

| | | |
|------|--|-----|
| I. | Bibliotecas del sistema Android. | 82 |
| II. | Bibliotecas Java. | 84 |
| III. | Versiones del sistema operativo Android. | 88 |
| IV. | Puertos y funciones del microcontrolador. | 146 |

| | | |
|-------|---|-----|
| V. | Programa del microcontrolador | 149 |
| VI. | Costos de materiales para el control..... | 174 |
| VII. | Costos de los sensores y actuadores. | 174 |
| VIII. | Costo de materiales para el cableado..... | 175 |
| IX. | Costo de fabricación e instalación del sistema..... | 175 |
| X. | Costo total del proyecto | 176 |

LISTA DE SÍMBOLOS

| | |
|----------------|---|
| APP | Aplicación |
| APK | <i>Application Package File</i> |
| BPS | Bits por segundo |
| CSMA/CA | <i>Carrier Sense Multiple Acces/Collision Detect</i> |
| RISC | Computador con conjunto de instrucciones reducidas |
| SWT | Conjunto de componentes para construir interfaces gráficas en Java |
| X86 | Conjunto de instrucciones utilizadas en la microarquitectura de CPU |
| PIC | Controlador interfaz periférico |
| H | Denota números hexadecimales |
| EHS | <i>Environmental Health and Safety</i> |
| Fosc | Frecuencia de oscilación |
| VDD | Fuente de alimentación positiva |
| Vss | Fuente de voltaje negativa |
| HZ | Hertz |
| API | Interfaz de programación de aplicaciones |
| kW | Kilovatios |
| SDK | Kit de desarrollo de software |
| NDK | Kit de desarrollo nativo |
| XML | Lenguaje de marcas extensible |
| C++ | Lenguaje de programación |
| MCLR | <i>Master Clear</i> |

| | |
|--------------|-------------------------------------|
| mA | Miliamperios |
| Ms | Milisegundos |
| PWM | Modulación por ancho de pulso |
| PCM | Modulación por impulsos codificados |
| OSC | Oscilador |
| RS232 | Protocolo de comunicación serial |
| EIB | Protocolo de comunicaciones de red |
| GND | Referencia de voltaje directo |
| SSR | Relé de estado sólido |
| PIR | Sensor infrarrojo pasivo |
| SMS | Servicio de mensajes cortos |
| VAC | Voltaje alterno |

GLOSARIO

| | |
|----------------------|--|
| Actuador | Elemento que convierte energía eléctrica en un movimiento mecánico. |
| Arrays | Un arreglo de espacios de memoria. |
| BatiBus | Protocolo de domótica totalmente abierto. |
| Baudios | Unidad de medida para los bits por segundo. |
| Bit | Señal electrónica que puede estar encendida (1) o apagada (0). |
| Bugs | Errores que se producen en un programa informático. |
| Bus | Caminos por los que viajan un grupo de datos. |
| Byte | Palabra lógica formada por 8 bits. |
| Cable coaxial | Cable que se utiliza para transmitir señales de electricidad de alta frecuencia. |
| Controlador | Elemento que realiza el trabajo de control en un sistema. |

| | |
|---------------------------|---|
| Cristal | Oscilador que vibra a una frecuencia establecida. |
| Entrada analógica | Entrada que recibe un voltaje variable de 0 a 5 V. |
| Fibra óptica | Cable conductor de la luz fabricado de vidrio o plástico flexible. |
| Flanco | Cambio de 0 a 1 lógico. |
| Fototransistor | Dispositivo semiconductor que permite conducir corriente debido a una luz. |
| <i>Framework</i> | Marco de aplicación o conjunto de bibliotecas orientadas para el desarrollo rápido de aplicaciones. |
| <i>Full-duplex</i> | Tipo de comunicación que permite transmitir y recibir datos al mismo tiempo. |
| Híbrido | Elemento formado por dos naturalezas distintas. |
| Infrarrojo | Emisor de ondas electromagnéticas, con luz no visible. |
| Interface | Puerto que permite enviar y recibir señales desde un componente a otro. |
| Interruptor Reed | Interruptor que se acciona mediante la aplicación de un campo magnético |

| | |
|-----------------------------|---|
| Java | Lenguaje de programación orientado a objetos. |
| Kernel | El núcleo o la parte fundamental de un sistema operativo. |
| Memoria volátil | Memoria que se borra cuando se desenergiza. |
| Oscilador | Generador de señales con diferentes frecuencias. |
| Pantalla táctil | Pantalla sensible al tacto. |
| Par apantallado | Tipo de cable que está recubierto por una malla metálica. |
| Par trenzado | Dos hilos de cobre aislados y entrelazados. |
| Pestillo | Pasador con que se asegura una puerta o una ventana. |
| <i>Plug and play</i> | La capacidad de un sistema informático de configurar automáticamente los dispositivos al conectarlos. |
| <i>Plugins</i> | Programa que puede anexarse a otro para aumentar sus funcionalidades. |
| Protocolo | Conjunto de normas y procedimientos útiles para la transmisión de datos conocido por el emisor y el receptor. |

| | |
|------------------------|--|
| <i>Qwerty</i> | Es una forma de ordenar las teclas en un teclado. |
| Radiofrecuencia | Las frecuencias del espectro electromagnético que se utilizan en las radio comunicaciones. |
| Semiconductor | Elemento que se comporta como un conductor o como un aislante. |
| Sensor | Elemento que detecta un fenómeno físico y lo convierte a una señal eléctrica. |
| Síncrono | Sistema de comunicación donde el transmisor y receptor se deben sincronizar. |
| Sistema binario | Sistema de numeración que utiliza solo cero y uno. |
| Termostato | Dispositivo que controla la temperatura. |
| Tiristor | Dispositivos semiconductores de potencia. |
| Trama | Paquete de datos que transmite una información. |
| Triac | Dispositivo electrónico de potencia. |
| <i>Watchdog</i> | Temporizador interno del microcontrolador. |
| <i>Widgets</i> | Una pequeña aplicación o programa. |

RESUMEN

En la automatización del hogar, el sector eléctrico comúnmente se conoce como domótica. Esta ciencia integra grupos de sistemas los cuales tienen la capacidad de brindar una gestión energética, seguridad y comunicación. En el trabajo de graduación que se presenta a continuación se presentan las bases para realizar el diseño de un sistema de automatización de una casa.

En el capítulo 1 se describen los tipos y modelos de sistemas domóticos que actualmente se pueden encontrar en el mercado. También se identifican las partes de la casa que se pueden automatizar, y los diferentes elementos que se pueden instalar en un sistema domótico.

El capítulo 2 describe el funcionamiento de los elementos del control principal, en un sistema domótico, como pueden ser los sensores, actuadores, el controlador del sistema y los medios de comunicación que va utilizar el usuario para comunicarse con el control.

En el capítulo 3 se describe el funcionamiento del sistema operativo Android. Los dispositivos que utilizan este sistema operativo y los tipos de software que sirven para realizar las aplicaciones Android.

En el capítulo 4 se describe el diseño de un control domótico utilizando un microcontrolador y un módulo de comunicación Bluetooth, la forma de instalar los actuadores que van a realizar las tareas de la casa y los sensores que van a brindar información al usuario, como también el diseño de la aplicación Android, utilizando la plataforma en línea APP Inventor.

OBJETIVOS

General

Diseñar un sistema de automatización para una casa, utilizando un microcontrolador PIC16F877A en comunicación Bluetooth con un celular con plataforma Android.

Específicos

1. Presentar los fundamentos de automatización de las funciones de una casa.
2. Dar a conocer los distintos elementos del control en un sistema de automatización de una casa.
3. Presentar los fundamentos del sistema Android y de la plataforma APP inventor.
4. Presentar el diseño del sistema de automatización.

INTRODUCCIÓN

En la actualidad, las condiciones en que se encuentran las casas, es buena, pero como todo lo que el ser humano construye, diseña o crea, estas han evolucionado debido a la búsqueda de la satisfacción del lugar en el que se pasa el mayor tiempo; esta satisfacción se da en diferentes rubros como confort, seguridad, ahorros de tiempo, esfuerzo y mayor diversión dentro de las mismas.

Las casas inteligentes dependen de un conjunto de tecnologías para poder subsistir; estos elementos se desarrollan dentro de la domótica la cual provee los recursos tecnológicos para estas primeras. Dentro de este conjunto de avances tecnológicos se encuentran los dispositivos comunes como actuadores, sensores y controladores, pero también los protocolos diseñados para trabajar con las casas inteligentes.

El poder definir los conceptos de domótica, casa inteligente, elementos y arquitectura ayuda a poder entender la manera en que se puede llevar a la práctica el desarrollo de un simulador de casas inteligentes; de esta manera se entiende que la domótica se construye a partir del término del latín *domus*, que significa casa, y del sufijo *otica*, referente a la automatización, es decir, la automatización de casas. Por lo tanto, se define a la domótica como la manera de crear casas, las cuales puedan ser autónomas para tomar decisiones acerca de la funcionalidad y administración de las mismas.

Es importante destacar que, las casas inteligentes son el conjunto de dispositivos externos e internos que se encuentran interconectados a una

computadora consiguiendo la fácil administración de todos los elementos instalados, proporcionando seguridad para la vida de los habitantes y confort dentro de la misma.

El teléfono celular ha dejado de ser un dispositivo dedicado para la comunicación, la mayoría de los celulares actuales incorporan reproductor de mp3, radio FM y además de ello, hasta el más simple de los teléfonos celulares incluye dentro de su sistema operativo aplicaciones de agenda, calendario, notas, memos, calculadora y alarma.

Los *smartphone* son complejos y ofrecen funciones que un teléfono celular básico no puede dar, con potentes procesadores de doble núcleo, muchos Mb de RAM e incluyen un sistema operativo casi tan potente como Windows, el cual puede ser Android, iOS o Windows Phone, entre otros.

Los teléfonos con sistema Android son uno de los más utilizados actualmente, una de las razones son las aplicaciones que se pueden descargar de la red o diseñar algún tipo de aplicación para solventar alguna necesidad. Una de las grandes empresas de la internet como lo es Google, ha diseñado una aplicación para automatizar una casa, Android Home. La idea de esta aplicación es poder controlar desde un celular la temperatura del hogar, las luces y otras actividades cotidianas.

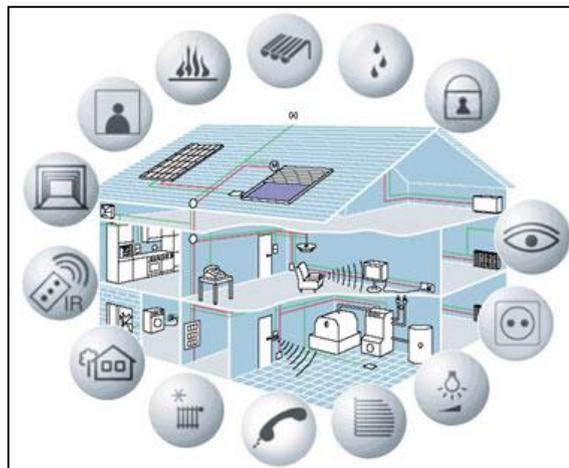
Actualmente ya existen algunas aplicaciones para automatizar las funciones de una casa pero todavía no se conocen o no se han desarrollado en Guatemala. Es por ello, que esta investigación persigue diseñar un sistema de automatización de una casa utilizando microcontrolador en comunicación Bluetooth con un teléfono celular con sistema Android, que sea eficiente y con un costo accesible.

1. AUTOMATIZACIÓN DE LAS FUNCIONES DE UNA CASA

La automatización del hogar empezó a difundirse en todos los niveles sociales y dejó de ser un privilegio reservado solo para personas de nivel socioeconómico alto. Estos procesos hacen más fácil la vida de las personas que habitan dichos hogares, permiten ahorros de energía importantes y mejorar los niveles de seguridad.

Una casa inteligente permite ayudar a disminuir el gasto energético para ahorrar dinero, y a su vez, cuidar el medio ambiente, brindar comodidad y tranquilidad cuando se está dentro o fuera de la casa, aumentar la seguridad, auxiliar y facilitar la organización de actividades cotidianas y realizar nuevas tareas desde casa.

Figura 1. Casa inteligente



Fuente: *Arqhys arquitectura*. <http://www.arqhys.com/construcciones/fotos/construcciones/Curso-de-domotica-gratis.jpg>. Consulta: 4 de marzo de 2015.

Al decir casa inteligente no significa que se controla por ella misma, sino que es más fácil controlar funciones como la iluminación, apertura de puertas y portones, control de persianas de ventanas, riego de jardín y control perimetral de la casa, por parte de las personas que viven en ella.

Las casas inteligentes dependen de un conjunto de tecnologías para poder subsistir; estos elementos se desarrollan dentro de la domótica la cual provee los recursos tecnológicos para éstas primeras. Dentro de este conjunto de avances tecnológicos se encuentran dispositivos comunes como son actuadores, sensores y controladores, y también los protocolos diseñados para trabajar con las casas inteligentes.

1.1. La domótica

El término domótica viene del latín *domus* que significa casa y de la palabra *otica* que significa automática, por lo tanto la domótica hace referencia a una casa automática o comúnmente llamada casa inteligente.

Por otro lado a la domótica se le describe como “la tecnología para desarrollar e implementar la automatización de instalaciones comunes en una casa o edificio. Los principales objetivos en la domótica son: seguridad personal y patrimonial en la vivienda, ahorro de energía, confort y comunicaciones”.

El concepto domótica se refiere a la automatización y control (encendido/apagado, apertura/cierre y regulación) de aparatos y sistemas de instalaciones eléctricas y electrotécnicos (iluminación, climatización, persianas y toldos, puertas y ventanas motorizados, el riego, entre otros) de forma centralizada y/o remota.

1.1.1. Elementos de un sistema domótico

La amplitud de una solución de domótica puede variar desde un único dispositivo, que realiza una sola acción, hasta amplios sistemas que controlan prácticamente todas las instalaciones dentro de la vivienda. Los distintos dispositivos de los sistemas de domótica se pueden clasificar en los siguientes grupos:

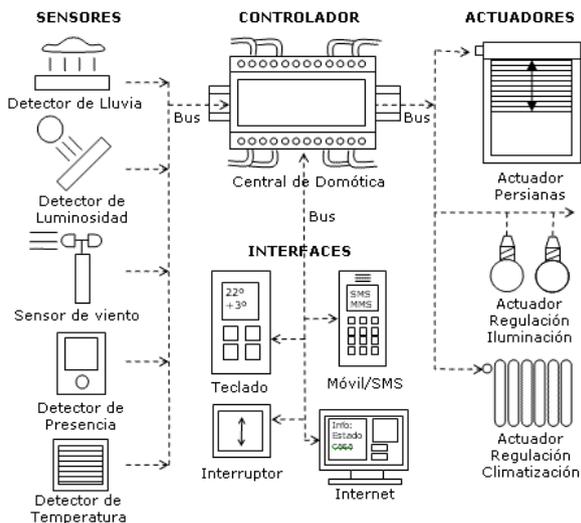
- **Controlador:** son los que permiten actuar sobre el sistema de una forma automática, por decisión tomada por centrales domóticas previamente programadas (que incluso puede ser una computadora personal), pulsadores, teclados, pantallas táctiles o no, mandos a distancia por infrarrojos IR (locales), por radiofrecuencia RF (hasta 50 metros), por teléfono, SMS o por PC (de forma local e incluso a través de internet).
- **Actuador:** es un dispositivo capaz de ejecutar y/o recibir una orden del controlador y realizar una acción sobre un aparato o sistema (encendido/apagado, subida/bajada, apertura/cierre, entre otros).
- **Sensor:** son los "ojos" o "la adquisición de datos" del sistema; pueden ser muy sofisticados, lo necesario es que lo pueda entender el sistema. Estos datos pueden ser órdenes directas a los actuadores o ir previamente a una central domótica y en función de la programación en ella introducida saldrá la orden final al actuador correspondiente. Ejemplos de sensores son los detectores de fuga de agua, de gas, de humo y/o fuego, de concentración de monóxido de carbono, de movimiento o intrusión y los termostatos.
- **Bus:** es el medio de transmisión que transporta la información entre los distintos dispositivos por un cableado propio, por la redes de otros

sistemas (red eléctrica, red telefónica, red de datos) o de forma inalámbrica.

- Interface: se refiere a los dispositivos (pantallas, móvil, internet y conectores) y los formatos (binario, audio) en que se muestra la información del sistema para los usuarios (u otros sistemas) y donde los mismos pueden interactuar con el sistema.

Es preciso destacar que todos los dispositivos del sistema de domótica no tienen que estar físicamente separados, sino varias funcionalidades pueden estar combinadas en un equipo. Por ejemplo, un equipo de central de domótica puede ser compuesto por un controlador, actuadores, sensores y varios interfaces.

Figura 2. **Elementos de un sistema de domótico**



Fuente: Casadomo soluciones. http://www.casadomo.com/images/news/canal_domotica_intro_ejemplos_dispositivos.gif. Consulta: 4 de marzo de 2015.

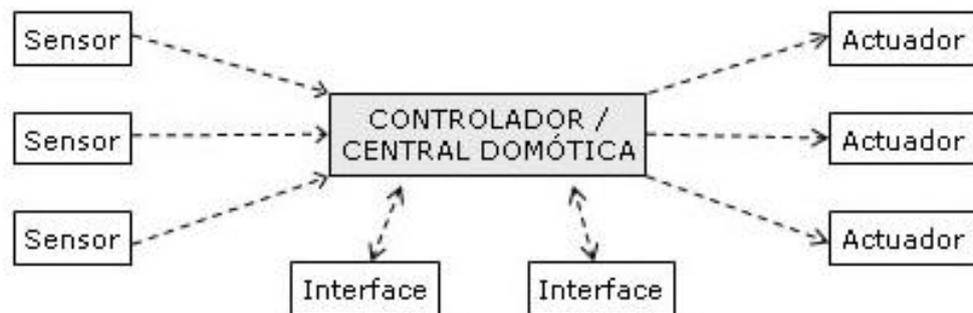
1.1.2. Arquitectura de los sistemas domóticos

La arquitectura de los sistemas de domótica hace referencia a la estructura de su red. La clasificación se realiza con base en donde reside la “inteligencia” del sistema domótico. Las principales arquitecturas son:

1.1.2.1. Arquitectura centralizada

En este tipo de arquitectura se tiene una topología de interconexión tipo estrella. Así, el sistema domótico posee un elemento de control central que es el encargado de manejar todas las señales de control de diversos dispositivos, y a su vez todos los dispositivos del sistema están conectados hacia él; por lo tanto si este elemento central falla o simplemente deja de funcionar, todo el sistema de control colapsa en su totalidad.

Figura 3. **Arquitectura domótica centralizada**



Fuente: *Casadomo soluciones*. http://www.casadomo.com/images/news/canal_domotica_intro_ejemplos_dispositivos.gif. Consulta: 4 de marzo de 2015.

1.1.2.2. Arquitectura descentralizada

En un sistema de domótica de arquitectura descentralizada, hay varios controladores, interconectados por un bus, que envían información entre ellos y a los actuadores e interfaces conectados a los controladores, según el programa, la configuración y la información que recibe de los sensores, sistemas interconectados y usuarios.

1.1.2.3. Arquitectura distribuida

Para esta arquitectura, el sistema de control se sitúa próximo al elemento a controlar, dando al sistema domótico gran flexibilidad, porque si uno de los dispositivos no puede ser controlado, no significa que los demás tampoco. Los factores más influyentes para la utilización de este tipo de arquitectura son los medios de transmisión, la velocidad en las comunicaciones y el tipo de protocolo a utilizar.

1.1.2.4. Arquitectura híbrida o mixta

En un sistema de domótica de arquitectura híbrida (también denominado arquitectura mixta) se combinan las arquitecturas de los sistemas, centralizadas, descentralizadas y distribuidas. A la vez que puede disponer de un controlador central o varios controladores descentralizados, los dispositivos de interfaces, sensores y actuadores pueden también ser controladores (como en un sistema “distribuido”) y procesar la información según el programa, la configuración, la información que capta por sí mismo, y tanto actuar como enviarla a otros dispositivos de la red, sin que necesariamente pase por otro controlador.

1.1.3. Medios de transmisión

El medio de transmisión de la información, interconexión y control, entre los distintos dispositivos de los sistemas de domótica puede ser de varios tipos. Los principales medios de transmisión son:

- Cableado propio: la transmisión por un cableado propio es el medio más común para los sistemas de domótica, principalmente son del tipo: par apantallado, par trenzado (1 a 4 pares), coaxial o fibra óptica.
- Cableado compartido: varias soluciones utilizan cables compartidos y/o redes existentes para la transmisión de su información, por ejemplo la red eléctrica (corrientes portadoras), la red telefónica o la red de datos.
- Inalámbrica: muchos sistemas de domótica utilizan soluciones de transmisión inalámbrica entre los distintos dispositivos, principalmente tecnologías de radiofrecuencia o infrarrojo.

Cuando el medio de transmisión está utilizado para transmitir información entre dispositivos con la función de “controlador” también se denomina “bus”. El bus también se utiliza muchas veces para alimentar a los dispositivos conectados a él (por ejemplo European Instalation *Bus* – EIB).

1.1.4. Protocolos de comunicación en la domótica

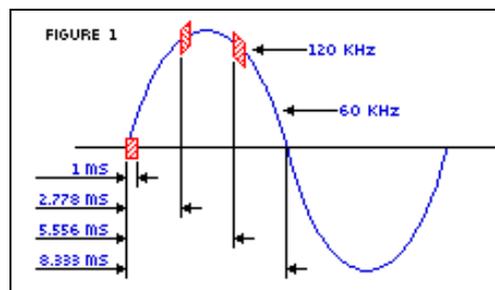
Partiendo de distintas arquitecturas que se presentan en los sistemas domóticos, se encuentran los diversos protocolos de comunicaciones, que son el idioma o formato de los mensajes que los diferentes elementos de control del sistema deben utilizar para entenderse unos con otros y para que puedan

intercambiar su información de una manera coherente. Algunos de los protocolos más importantes orientados hacia la automatización de casas se describen a continuación.

1.1.4.1. X-10

El protocolo X-10 es aquel orientado hacia la utilización de la red eléctrica de las viviendas. Allí se utilizan corrientes portadoras para controlar cualquier dispositivo a través de la línea de corriente doméstica (120 o 220 voltios y 50 o 60 Hz), y se hace modulando impulsos de 120 KHz. Con este protocolo se maneja un direccionamiento sencillo que se puede utilizar en la red para identificar cualquier elemento.

Figura 4. Protocolo X-10



Fuente: *Tecnología 2.0*. <https://tecnologiadospuncocero.wordpress.com/category/domotica/>.

Consulta: 4 de marzo de 2015.

Con este protocolo se maneja un direccionamiento sencillo que se puede utilizar en la red para identificar cualquier elemento. El protocolo tiene un margen de acción que contempla 16 grupos de direcciones llamados *housecodes* y 16 direcciones individuales llamadas *unit codes*.

Este protocolo posee tramas de datos que son ceros y agrupados formando comandos; con esta agrupación se pueden formar hasta seis acciones para el dispositivo que son: encendido, apagado, reducir, aumentar, todo encendido y todo apagado.

Las señales que se envían son recibidas por todos los módulos, pero de acuerdo con la dirección solo actúan sobre el dispositivo al que va dirigida la acción (los primeros bits de la señal son el identificador del módulo, de este modo se sabe dónde está el dispositivo a controlar).

Las características principales de X10:

- Es estándar debido a las características de la corriente doméstica (120 o 220 V. y 50 o 60 Hz.).
- No hay que configurar nada para que entre en funcionamiento (*plug and play*).
- Es una tecnología que aprovecha la red eléctrica que ya está instalada en la vivienda.
- Modularidad y capacidad de crecimiento, con componentes fáciles de instalar y que no requieren cableados especiales.
- Capacidad de interfuncionamiento entre productos.

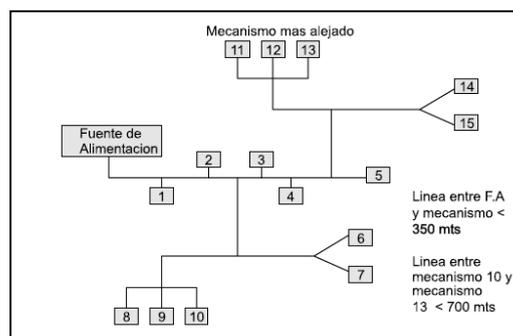
El protocolo X-10 no es propietario, es decir, que cualquier fabricante puede producir dispositivos X-10 y venderlos, pero está obligado a usar circuitos de control que han sido diseñados por el fabricante de esta tecnología.

1.1.4.2. EIB (European Installation Bus)

Es un sistema domótico desarrollado bajo el aval y supervisión de la unión europea con el objetivo de evitar las importaciones de productos similares que se estaban produciendo en el mercado japonés y norteamericano, donde esta clase de tecnología se ha desarrollado antes que en Europa.

Este estándar europeo define la relación extremo a extremo entre los dispositivos, permitiendo distribuir la inteligencia entre los sensores y los actuadores instalados en la vivienda. Una arquitectura EIB puede llegar a tener 11,520 componentes conectados al mismo bus; el sistema EIB permite, mediante la instalación de algunos amplificadores o repetidores, optimizar su sistema hasta de 256 dispositivos por línea, pero para evitar colisiones entre telegramas enviados se utiliza la técnica CSMA/CA (*Carrier Sense Multiple Access*), garantizando el acceso libre de colisiones fortuitas en la línea bus y sin ninguna pérdida de datos.

Figura 5. **Arquitectura del sistema EIB**



Fuente: *Casadomo soluciones*. http://www.casadomo.com/images/news/canal_domotica_intro_ejemplos_dispositivos.gif. Consulta: 4 de marzo de 2015.

1.1.4.3. Konnex

Esta es la iniciativa de tres asociaciones europeas (EIBA, EHSA y BCI) con el objetivo de unificar los esfuerzos de muchos fabricantes de sistemas domóticos del mercado europeo. El modelo *Konnex* es el paso evolutivo lógico que trata de concentrar toda la experiencia y conocimientos de los principales estándares europeos EIB, EHS, BatiBus (modelo que de acuerdo con la convergencia está ya dentro de Konnex).

Este sistema es impulsado altamente para competir contra el mercado norteamericano que se encuentra en dominio de Lonworks. El CENELEC (*European Comité for Electrotechnical Standarization*) lo ha aprobado con la norma EN-50090. En sí el modelo Konnex se basa en el modelo EIB y expande su funcionalidad añadiendo nuevos medios físicos y modos de configuración BatiBus y EHS. Contempla tres modos de trabajo que pueden seleccionarse dependiendo del nivel de competencia de cada instalador:

- Modo S. (modo sistema): este modelo sigue la misma configuración que el EIB actual, es decir, los diversos dispositivos o nidos de la nueva instalación son ejecutados por profesionales con ayuda de *software*.
- Modo E. (modo fácil): acá los dispositivos son programados en fábrica para realizar una función concreta. También requieren alguna configuración por parte de un controlador central.
- Modo A. (modo automático): se sigue una filosofía *plug and play*, es decir que no se tiene que configurar el dispositivo.

Por otro lado Konnex puede funcionar por cuatro medios físicos distintos:

- Par trenzado (TP1 y TP0)
- Corrientes portadoras (PL100 y PL132)
- Ethernet (IP)
- Radiofrecuencia (RF)

La posibilidad de utilizar medios físicos distintos permite a los instaladores adaptar la red a condiciones más favorables para el usuario final. Konnex está basado en el modelo de EIB, siendo compatible con los distintos productos elaborados por varias de las firmas productoras de componentes EIB.

1.1.4.4. Lonworks

Es un sistema de control domótico propietario presentada por la firma Echelon en 1992. Debido a su costo los dispositivos *lonworks* no han tenido una implantación masiva en los hogares. *Lonworks* es un sistema americano que está más implantado en Estados Unidos que en Europa.

Los fabricantes de dispositivos *lonworks* deben emplear en cada uno de sus dispositivos un microcontrolador especial que le denominan *neuron chip*. Este circuito integrado al igual que el *firmware* que implementa el protocolo Lontalk utilizado en los nodos *lonworks* para comunicarse, fueron desarrollados por ECHELON en 1990. Una ventaja importante de *lonworks* respecto de otras tecnologías es que implementa el modelo de referencia OSI. De este modo, servicios tales como el reenvío automático tras una pérdida de trama o la autenticación del emisor de la trama, están completamente implementadas en la solución *lonworks*.

1.1.4.5. JINI

Esta tecnología de Sun Microsystems permite descubrir nuevos dispositivos que se van incorporando a la red del hogar mediante cualquier medio. JINI permite utilizar servicios y dispositivos de red de manera tan fácil como utilizar una conexión telefónica, es decir, permite conectarse y participar por medio de un tono de marcado en red. La meta de JINI es simplificar interacción en la red.

JINI aprovecha la tecnología JAVA y consiste en una pequeña cantidad de este código en forma de librerías de clases y algunas convenciones para crear una "federación" de máquinas JAVA virtuales en la red, similar a la creación de una comunidad. Los usuarios navegantes de la red, tales como usuarios, dispositivos, datos, y aplicaciones, se conectan dinámicamente para compartir información y realizar tareas, convirtiéndose en una unidad lógica de información de red.

Cada aparato de esta red tiene en principio dos conexiones: el enchufe a la red eléctrica y una ficha de conexión del tipo RJ-45. Está diseñado para ejecutarse en periféricos, pero puede hacerlo en potentes ordenadores, cámaras, teléfonos o cualquier dispositivo electrónico.

1.1.4.6. TCP/IP

La mayoría de los protocolos domóticos han sido especialmente creados para implementar redes de control distribuidas (*Lonworks*, EIB y X-10), las tramas fueron diseñadas de forma que el espacio útil para datos de las aplicaciones fuera el máximo. Por ejemplo para encender y apagar una luz basta con una orden codificada en un par de octetos. Por lo tanto, se trata de

minimizar los campos de control que el protocolo necesita para transferir estos dos octetos al dispositivo destino.

TCP/IP está siendo usado en infinidad de computadoras y aplicaciones, de forma que ha conseguido un volumen de negocio, tal que ha hecho de este protocolo la herramienta ideal para asegurar la interconectividad total entre máquinas en cualquier parte del mundo y con esto se puede hablar cada vez mas de viviendas inteligentes.

1.1.5. Ventajas de la domótica

En una vivienda, oficina o grandes edificios como hoteles y espacios públicos, la domótica ofrece ventajas adaptadas a las necesidades reales. Entre las principales ventajas que proporciona la domótica están:

1.1.5.1. Confort

La domótica aporta una serie de comodidades, que redundan en una mejor gestión del entorno (tanto personal como profesional), consiguiendo un ambiente más confortable y habitable.

- Iluminación y climatización: encendido y apagado de los sistemas de iluminación y climatización por medio de sensores de presencia y temperatura; regulación de la intensidad luminosa en función de la cantidad de luz ambiental existente y grabación y reproducción de escenas de iluminación.
- Persianas y toldos: control manual (mediante pulsadores) y/o control automático, dependiendo de la climatología o de un horario establecido. El

sistema automático actuará, por ejemplo, si está lloviendo. En ese caso se recogen los toldos y se bajan las persianas. Si hace sol, se bajan los toldos y se suben las persianas. Si hay viento, se suben los toldos.

- Enchufes: control de los aparatos conectados a los enchufes mediante el corte del suministro eléctrico de los mismos. Este sistema puede ser muy eficaz para evitar que los niños sufran descargas eléctricas al introducir los dedos o algún objeto en los enchufes o desconectar o conectar los enchufes conectados a aparatos de televisión, horno, ventiladores, radiadores, entre otros. Este control también puede hacerse remotamente a través del teléfono o de internet.
- Visualización y control: conocimiento en tiempo real del estado de todos los elementos de la instalación domótica, a través de pantallas informativas o táctiles, y/o a través de internet/intranet. En función del estado del sistema, se podrán controlar los elementos deseados. Asimismo se podrán controlar los elementos del sistema a través del teléfono.

1.1.5.2. Seguridad

La domótica permite integrar las diferentes áreas de seguridad existentes y gestionarlas de una manera inteligente, proporcionando mayor protección y tranquilidad al usuario.

- Incidentes y averías: mediante sensores se pueden detectar los incendios, humo, fugas de gas y agua y mediante el módulo telefónico se puede desviar la alarma a un número predeterminado (número propio y servicios de urgencia).

- Seguridad de los bienes: gestión del control de acceso y control de movimientos, así como la simulación de presencia y alarmas antiintrusión.
- Alarmas técnicas: mediante sensores se pueden detectar incendios, fugas de gas y agua. En caso de detectar alguna de estas eventualidades, la instalación corta el suministro y avisa por teléfono a un número prefijado por el usuario. Para llevar a cabo este control, existen sensores de humo o fuego, gas e inundación.
- Alarmas médicas: este tipo de alarmas son de especial utilidad para las personas mayores y los enfermos. Por medio de un pulsador de radiofrecuencia que accione el usuario, el sistema puede avisar a los servicios de urgencia, como ambulancias y policía.
- Intrusión y video vigilancia: detección de intrusos por medio de sensores y cámaras de video IP, lo que permite visualizar el vídeo en tiempo real a través de internet y dar aviso a un número de teléfono prefijado, en caso de alarma.
- Simulación de presencia: programación del encendido y apagado de luces para simular actividad en el edificio o vivienda, así como apertura y cierre de persianas.
- Control de accesos: gestión de accesos a edificios por medio de lectores de identificación (lectores de tarjetas con banda magnética/chip, lectores biométricos), cerrojos electrónicos y otros dispositivos. Utilizando un software de control se consigue llevar un control exhaustivo de toda la información (accesos según horarios, visitas, presencia, alarmas y estadísticas).

1.1.5.3. Ahorro energético

El diseño de una instalación domótica está enfocado en gran medida hacia el ahorro de consumo energético y el aprovechamiento óptimo, económico y racional de la energía necesaria para el funcionamiento del edificio

Se puede gestionar eficazmente y de manera óptima el consumo energético de las instalaciones eléctricas o de combustibles fósiles, lo que contribuye a la protección del medio ambiente y permite reducir significativamente el consumo energético. Abacotech S. L. ofrece las siguientes características para potenciar el ahorro energético:

- **Iluminación:** mediante una simple pulsación, el usuario puede apagar una zona o todas las luces del edificio o vivienda. Por otra parte, los detectores de movimiento consiguen que las luces no estén encendidas más tiempo del necesario. En el caso del alumbrado exterior, los sensores de iluminación registrarán el encendido o apagado del mismo. Asimismo, la regulación constante de la iluminación en zonas de mayor tránsito evita el ciclo de encendido y apagado de la iluminación fluorescente. Es posible lograr un ahorro de energía de hasta un 75 % en iluminación, llevando a cabo un control de luz por medio de sensores de luminosidad y presencia.
- **Temperatura / climatización:** mediante el control individualizado por zonas y estableciendo perfiles de temperatura individuales, se puede alcanzar un ahorro de energía de más de un 30 % al año. De la misma manera, el sistema permite acondicionar aquellas zonas que se estén utilizando y dejar el resto a una temperatura de ahorro. Una gestión inteligente de persianas y toldos ayuda a reducir sensiblemente el consumo de la climatización.

- Grifería: la grifería domótica promueve el ahorro del consumo de agua, empleando solo la cantidad necesaria de la misma. Para ello, utiliza los siguientes mecanismos:
 - Detección de presencia: los grifos se abren y cierran por medio de una electroválvula conectada a un sensor.
 - Temporizador: el agua deja de salir cuando vuelve a pasar la mano por delante del mismo o cuando transcurre la duración programada.
 - Control de temperatura: se consigue la temperatura deseada de manera precisa, sin necesidad de hacer una regulación manual, con el gasto de agua adicional que esta regulación supone.
 - Contador de consumo de agua: limitación de caudal máximo (esta función no es programable a través de EIB).

1.1.6. Desventajas de la domótica

Existen algunas debilidades al implementar la domótica en un hogar, debido fundamentalmente a lo nuevo de la tecnología y, por lo tanto, la inexperiencia en la entrega de los servicios. Dentro de estas debilidades están algunos puntos que hay que considerar:

- Alto precio de los aparatos domóticos (instalación y equipos), aunque se espera que con el tiempo estos valores disminuyan debido a la competencia entre empresas.
- Uso de internet (fundamental en el uso de la domótica) hace que el sistema sea vulnerable a accesos no autorizados y provocar una inestabilidad en dicho sistema.

- El usuario puede no percibir como un valor añadido un sistema totalmente centralizado que controle todos los aspectos de una casa. Existen personas que prefieren tener controles individuales, como un sistema independiente de control de la calefacción, un sistema de alarma, entre otros.
- El costo adicional de la instalación de un sistema domótico supone para muchas personas un gasto superfluo.
- Si se decide utilizar un sistema propietario, el usuario estará obligado a utilizar siempre los dispositivos del mismo fabricante, lo que supone una desventaja.
- Si se produce un desperfecto del sistema de control central, puede ocurrir que algunos aparatos conectados a dicho sistema dejen de ser operativos incluso de forma manual.
- En caso de averías puede suceder que el usuario no sepa a qué servicio técnico llamar, si al servicio técnico de la red domótica o al del aparato electrodoméstico averiado.
- Además de los dispositivos que ofrece el fabricante de aparatos domóticos para su sistema, también pueden existir en el mercado, electrodomésticos de mayor calidad que no puedan utilizarse en el sistema domótico, por no ser compatibles con el mismo.
- Es muy importante que todos los dispositivos de seguridad instalados, sobre todo los encargados de cortar el gas, funcionen correctamente, ya

que una instalación incorrecta de los mismos o su malfuncionamiento pueden resultar muy peligrosos.

- Disponer de una red domótica implica, como en cualquier red, la aparición de problemas de funcionamiento o desperfectos que pueden causar importantes inconvenientes para el usuario.

1.1.7. Identificación de las funciones para automatizar una casa

Al querer automatizar las funciones de una casa se deben tomar en consideración las necesidades principales que tiene el usuario. Los sistemas domóticos pueden ser implementados tanto en la vivienda existente, como en la de nueva construcción, siendo en este último caso la implementación evidentemente más barata. La oferta actual de productos y servicios domóticos es atractiva y se adapta a cualquier topología de vivienda o edificio (viviendas particulares, oficinas, hoteles, hospitales, colegios, universidades, entre otros), ya sea existente o de nueva construcción.

En el caso de viviendas de nueva construcción o de reconstrucción profunda, se recomienda la colocación de un cableado específico que transmita la información necesaria entre los diferentes elementos del sistema. La normativa actual obliga a todos los edificios de nueva construcción a contar con unas instalaciones mínimas de telecomunicaciones, que pueden ampliarse fácilmente para albergar sistemas domóticos que permitan la automatización de funciones en el hogar. En el caso de viviendas existentes, los requisitos de instalación son mínimos, ya que es posible aprovechar la red eléctrica de la vivienda o utilizar tecnologías inalámbricas como medio de transmisión. En

cualquier caso, los mecanismos de uso son actualmente sencillos y su costo ha disminuido en los últimos años.

Para cualquiera de los casos, se debe disponer de un plano de la casa, para analizar las funciones que se desean automatizar como: el medio de comunicación ya sea cableada o inalámbrica, la instalación de sensores, actuadores, los controles principales y la interface que va a tener el usuario con los controles de la casa.

Figura 6. **Plano de una casa de un nivel**



Fuente: elaboración propia, empleando AutoCAD 2008.

En el plano de una casa se deben tomar en cuenta los niveles que posee, los dormitorios, baños, cocina y sala, parqueo, jardín, puertas, ventanas, área perimetral, la instalación eléctrica y de agua; para analizar cuáles serán las áreas o funciones de la casa que se van automatizar.

1.1.7.1. Control de iluminación

La iluminación inteligente y los sistemas de control de iluminación en muchas ocasiones son consideradas como las partes más importantes de la domótica. La iluminación de un espacio puede realizar cambios dramáticos en el ambiente del lugar y en el ahorro de energía como: encender o apagar las luces o graduar la luminosidad, ya sea de forma automática o a criterio del usuario mediante una interface, y tener programados escenarios que simulen una casa habitada, son los principales objetivos de automatizar la iluminación.

Existen varios tipos de lámparas que cubren diferentes necesidades según las aplicaciones y condiciones de uso que, en cada caso, se quieran cubrir. Es muy importante desde el punto de vista ecológico saber qué fuente de luz elegir, ya que, de esta manera se evita el gasto innecesario de energía, lo que supone un acierto desde el punto de vista económico y ecológico. Cada vez es más frecuente que las lámparas ofrezcan mayor cantidad de luz, mejor reproducción cromática y un menor consumo energético, lo que conlleva un mayor respeto al medio ambiente.

1.1.7.1.1. Tipos de lámparas

- La lámpara incandescente: esta produce luz por medio del calentamiento eléctrico de un alambre (el filamento de tungsteno se pone incandescente cuando pasa corriente por él) a una temperatura alta que la radiación

emite en el campo visible del espectro. Son las más antiguas fuentes de luz conocidas con las que se obtiene la mejor reproducción de los colores, muy cercana a la luz natural del sol. Su desventaja es la corta vida de funcionamiento, baja eficacia luminosa (ya que el 90 % de la energía se pierde en forma de calor) y depreciación luminosa respecto del tiempo. La ventaja es que tienen un costo de adquisición bajo y su instalación resulta simple, al no necesitar de equipos auxiliares.

Figura 7. **Lámpara incandescente, inducción electromagnética y led**



Fuente: elaboración propia, empleando Paint.

- Lámparas de inducción electromagnética: son las más utilizadas actualmente. Su interior está relleno de vapor de mercurio a baja presión que, al contacto con una descarga eléctrica, produce luz ultravioleta. A su vez, esta luz, al contacto con el polvo fluorescente que recubre el interior del tubo, produce luz normal, prescindiendo de electrodos para originar la ionización. Estos electrodos se sustituyen por una bobina de inducción sin filamentos y una antena acopladora (cuya potencia proviene de un generador externo de alta frecuencia, habitualmente 2,65 MHz). Ambos elementos crean un campo electromagnético que introduce la corriente eléctrica en el gas, provocando su ionización. En el mercado hay multitud de casquillos comunes que las hacen compatibles con casi cualquier

luminaria. De este tipo de lámpara existen con varias formas (de espiral o de tubo) y tipos de encendido (electrónico y eléctrico). La ventaja principal es el gran incremento de la vida de la lámpara, entre 20.000 y 100.000 h. y el ahorro energético debido que a bajas potencias se logra conseguir una alta luminosidad.

- Lámparas led: la tecnología led basada en SSL (*solid state lighting*) incorpora las grandes ventajas de los dispositivos electrónicos, duración y fiabilidad, permite manejar un voltaje bajo, entre 1 y 4 voltios. La corriente eléctrica se encuentra en un rango entre 10 y 40 miliamperios. Los led están formados por un diodo semiconductor con 2 regiones claramente diferenciadas (zona p para cargas positivas y zona n para negativas), utilizando varios formatos de encapsulamiento en chip (DIP, SMD, COB, entre otros). Se pueden fabricar leds que proporcionen cualquier color visible, permitiendo una variedad cromática muy amplia (excepto una determinada gama de amarillo, ya que no se conoce ningún semiconductor que genere una longitud de onda entre los 550 y los 585 nm).

1.1.7.2. Apertura de puertas

La motorización e integración de las puertas y cerraduras electrónicas dentro del control domótico trata tanto de facilitar el acceso a la vivienda del propio usuario, como permitir el acceso (abrir, cerrar), a otras personas que necesiten acceder a la misma (visitas, mensajeros, entre otros.). Para las puertas de acceso que demandan apertura con llave, existen llaves electrónicas de aproximación que logran que el usuario pueda abrir, simplemente acercándose a la puerta con la llave en el bolsillo.

Desde dentro de la vivienda, con el interfaz que el mismo usuario se sienta más cómodo (mando a distancia, teléfono, entre otros). También se pueden abrir las puertas de acceso a la vivienda, para dejar entrar a visitas de forma remota sin tener que acercarse a la puerta. Las puertas motorizadas dentro de la casa también se pueden controlar y abrir con gran variedad de interfaces de forma activa, por ejemplo, con un mando a distancia, o de forma pasiva, con un detector de movimiento.

1.1.7.2.1. Tipos de cerraduras

- La cerradura magnética: dispositivo de fijación simple que consiste en una placa del electroimán y de la armadura. Atando el electroimán al marco de puerta y la placa de la armadura a la puerta, un paso actual a través del electroimán atrae la placa de la armadura que sostiene la puerta cerrada. A diferencia de una cerradura eléctrica una cerradura magnética no tiene ninguna pieza de interconexión y es por lo tanto no conveniente para los usos de la alta seguridad, porque es posible interrumpir la fuente de alimentación dejando libre la cerradura.
- Cerradura de pestillo: este destrabador realiza la apertura de una cerradura estándar liberando el pestillo en forma eléctrica, comandado por los controles de accesos. Es el tipo de cerradura más común y económico, que ofrece un grado de seguridad bajo. Es ideal para instalaciones simples donde se utiliza la cerradura existente. Se instala sobre el marco de la puerta, por lo general embutido. Los hay de dos tipos de activación:
 - Activación con energía: para la apertura necesita ser energizado y traba la puerta sin necesidad de corriente eléctrica.

- Activación sin energía: para la apertura debe quitarse la energía, ya que la traba la realiza estando energizado.

La ventaja de este destrabador es que puede utilizarse para casos de emergencia donde hay que asegurar la apertura sin disponer del resto de la instalación, ya que se le puede quitar la energía con un pulsador de corte de emergencia. Su calidad y costo son mayores.

- Cerradura motorizada: estas cerraduras son las ideales para áreas sensibles donde la seguridad es fundamental. Trabajan como una cerradura de doble perno que traba siempre la puerta al cerrarse; no requiere energía para el cierre y brinda la comodidad de una apertura mediante llaves computadas para el caso de emergencia. La apertura se realiza a través de un micromotor de baja tensión comandado por el control de accesos, con una operación silenciosa y rápida. La instalación se realiza como una cerradura estándar de embutir dentro de la hoja de la puerta, con las medidas de una cerradura común. Tiene asociada una fuente de alimentación que permite su uso como UPS, agregándole una batería pequeña. Además se le puede accionar fácilmente desde un portero eléctrico, un pulsador o una central telefónica, ya que posee entradas especialmente diseñadas para facilitar la conexión a estos dispositivos.
- Cerraduras eléctricas: ideales para utilizar en portones o puertas pesadas. Por su forma física, resisten muy bien la fuerza y la inercia. Se montan directamente sobre el portón y no se requiere de hacer adaptaciones de ningún tipo para embutirla. En síntesis, se trata de un pestillo grande reforzado que, como ya se dijo, se adapta a estas aplicaciones más

pesadas. En caso de corte de energía, poseen una llave mecánica que permite abrirlas manualmente.

Figura 8. **Cerradura electromagnética, pestillo, motorizada y eléctrica**



Fuente: elaboración propia, empleando Paint.

1.1.7.3. **Portón eléctrico**

Los portones eléctricos representan una forma conveniente de aumentar la seguridad, tanto en el hogar como en el comercio, así como también proporcionar mayor comodidad. Se basan en la utilización de un motor con una serie de engranajes y partes mecánicas móviles, que hacen que el portón se abra o cierre. Dependiendo del tipo de movimiento del portón se encuentran distintos tipos:

- **Seccionales o enrollables:** su mecanismo es similar al de una persiana enrollable de una ventana pero a una escala mayor, aunque también pueden no ser enrollables sino que se deslizan seccionalmente hacia arriba y atrás (otra opción son los plegables, generalmente con un pliegue). Son la opción ideal en caso de garajes para dos automóviles. Estos portones son adecuados para parqueos estrechos, tienen una gran estética y son sencillos de instalar y usar; no ocupan espacio exterior al marco al abrir y cerrar y no requieren de mayor mantenimiento.

- Basculantes o de contrapeso: los portones levadizos contrapesados o de contrapeso tienen las mismas ventajas y desventajas que los seccionales, maximizan el espacio lateral pero minimizan el superior. A diferencia de los seccionales estos pueden tener adosada una puerta en la totalidad de los casos, tener o no saliente (el portón queda en parte hacia el exterior al abrir) de acuerdo con la ubicación que se tenga del parqueo y espacio público y se balancean a partir de contrapesos.
- Corredizos: son usados tanto en viviendas como en sectores industriales. Maximizan el espacio superior y tienen la desventaja de quitar espacio lateral (además de necesitar, cuando se deslizan sobre una guía, una superficie horizontal para su movimiento). Se pueden utilizar tanto en parqueos abiertos como cerrados y pueden ser rectos o seccionales como los levadizos, dependiendo del espacio con el que se cuente. También hay portones corredizos dobles en caso de parqueos para dos automóviles (donde un panel se corre tras el otro para permitir la salida o entrada de un vehículo a la vez al estacionamiento).

Figura 9. **Portón corredizo**



Fuente: *PFC Controls*. http://www.pfccontrols.com/lang1/gallery/lak_dcross_011.jpg. Consulta: 4 de marzo de 2015.

- Abatibles o rebatibles: también llamados abatientes o pivotantes. Son muy comunes entre los portones manuales y cada vez más en los automáticos. Son silenciosos, fáciles de instalar y mantener. Habitualmente se les utiliza con rejas pero pueden ser contruidos en cualquier material, es por esto que son los predilectos a la hora de trabajos de herrería y decoración de entradas residenciales. Puede optarse por abertura de una o dos hojas y su abertura pueden ser hacia el interior o exterior.

1.1.7.4. Apertura de ventanas

Las ventanas motorizadas integradas con el sistema de domótica permiten automatizar tareas de ventilación (para renovar el aire, refrigerar, calentar o evacuar humo) que conlleva un mejor confort interior y ahorro energético. Las tareas de ventilación pueden ser programadas para garantizar un tiempo óptimo de apertura de ventilación, según la situación climatológica exterior y el ambiente interior preferido.

1.1.7.4.1. Apertura de persianas de ventanas

La subida y bajada de las persianas motorizadas son tradicionalmente controladas directamente a través de pulsadores de pared. Con las persianas motorizadas integradas en un sistema de domótica, se puede mejorar el ahorro energético, aumentar el confort y mejorar la seguridad dentro y fuera del hogar.

Se puede controlar la subida y bajada de las persianas motorizadas desde un sistema de domótica, de forma centralizada o remota, por ejemplo, mediante una programación horaria, el uso de sensores que detectan las condiciones meteorológicas, la intensidad de la luz natural o el estado de alarma de

intrusión. El control y automatización de las persianas motorizadas se realiza en uno o varios grupos, normalmente divididos por estancias.

La luz natural puede controlar la subida y bajada de las persianas para poder aprovechar o evitar su entrada en la vivienda: por ejemplo se puede dejar entrar el sol para iluminar y calentar la casa en las temporadas de frío y oscuridad en el invierno, apoyando a los sistemas de calefacción y la iluminación artificial; o se puede evitar que entre el sol en las épocas de calor, para que no caliente la casa.

Figura 10. **Persiana motorizada para ventana**



Fuente: *Domotic*. <http://2.bp.blogspot.com/-X2XqzocZCmU/TcGGJ4oUKOI/AAAAAADR4/LMeKOK-nMtA/s1600/1.jpg>. Consulta: 4 de marzo de 2015.

1.1.7.5. Regado de jardín

El riego automático es una aplicación muy utilizada en viviendas unifamiliares y en las comunidades de vecinos, que ahorra tiempo, agua y mejora la calidad del riego en comparación con la apertura manual de las llaves de agua. La automatización y control del riego automático con el sistema de domótica, o un programador avanzado, permite no solo basar el riego en la temporización, sino tener el riego automático totalmente controlado, tanto de

forma local como remota. El sistema de riego se basa en la distribución de emisores (aspersores, difusores, tubos de goteo, entre otros) por el jardín (conectados a la alimentación del agua) según la necesidad de riego de cada zona. La estructura de la instalación del riego se divide en sectores según lo que permite la presión y el caudal de la toma principal de agua. Los sistemas de riego automático disponen de los siguientes componentes:

- Electroválvulas
- Arquetas
- Tuberías
- Reductor de presión
- Emisores de riego
- Sensores

Figura 11. **Sistema de riego eléctrico**



Fuente: Domotek. http://www.domotek.com.mx/Domotica/imagenes_catalogo/f6cd7d_31701.jpg.

Consulta: 4 de marzo de 2015.

1.1.7.6. Control perimetral de la casa

El sistema domótico puede controlar intrusiones tanto exteriores como interiores, con sensores destinados a encender las luces en las zonas de paso

de forma automática, mandar una señal al usuario o generar una alarma sonora; estas últimas pueden configurarse para la detección de intrusos. Se pueden instalar diferentes tipos de sensores, como por ejemplo: volumétricos, sensores de movimiento o presencia, sensores de apertura en puertas y ventanas. En caso de alarma se puede programar al sistema para que realice toda una serie de llamadas telefónicas para verificar el estado, o simplemente la alarma puede enviarse a la central de alarmas, que será la encargada de llamar al usuario o avisar a la policía en caso necesario.

1.1.7.6.1. Sensores de movimiento

El detector de movimiento RF es un detector inalámbrico con tecnología PIR para detectar intrusos en interiores. Esta tecnología detecta cambios bruscos de temperatura. Con base en la tecnología de infrarrojos pasiva, un sensor de movimiento detecta el movimiento mediante la detección de la luz infrarroja, o calor, emitida por una persona. Una vez que este calor se ha detectado, el sensor de movimiento lo comunicará al sistema.

Además de los sensores de movimiento infrarrojos estándar, están también disponibles los sensores activos de movimientos infrarrojos que utilizan un diodo emisor de luz y un fototransistor. Otro tipo de detector de movimiento es el sensor de movimiento por ultrasonidos, que lee el movimiento mediante el uso de ondas ultrasónicas.

Los sensores de movimiento pueden ser usados para moderar el uso de aparatos domésticos como la iluminación interior y exterior, cámaras, aspersores y los interruptores de luz. Los detectores de movimiento se utilizan con frecuencia para fines de seguridad, como para activar la iluminación, puertas, cerraduras y alarmas.

Figura 12. **Sensor de movimiento**



Fuente: *Robert Next*. http://www.robertnext.com.ar/prod_images/0000000055_1.jpg. Consulta: 4 de marzo de 2015.

1.1.7.6.2. Sensores magnéticos

Detecta los campos magnéticos que provocan los imanes o las corrientes eléctricas. El principal es el llamado interruptor *Reed*; el cual consiste en un par de láminas metálicas de materiales ferromagnéticos metidas en el interior de una cápsula que se atraen en presencia de un campo magnético, cerrando el circuito. El interruptor *Reed* puede sustituir a los finales de carrera para detectar la posición de un elemento móvil, con la ventaja de que no necesita ser empujado físicamente por dicho elemento, sino que puede detectar la proximidad sin contacto directo. Esto es muy útil cuando interesa evitar el contacto físico, por ejemplo para detectar el nivel de agua de un depósito sin riesgo de cortocircuitos.

Este sensor magnético se suele instalar en las puertas exteriores y ventanas que son fácilmente accesibles. Los contactos funcionan según el principio de que cuando la puerta o ventana se mueve más que 1/2 pulgada, el campo magnético en el conmutador se retira y se rompe el circuito de alarma (o zona).

Una vez disparado, la señal de alarma no se puede detener con solo cerrar la puerta o ventana. Los sensores magnéticos son populares y ampliamente utilizados debido a que detectan a un intruso antes de que entre en las instalaciones protegidas.

Figura 13. **Sensores magnéticos para puerta o ventana**



Fuente: *DIGEL Eléctrica*. <http://www.digel.com.br>

http://digel.com.br/novosite/components/com_virtuemart/shop_image/product/Sensores_magn_t_4d5a8311bf4b1.jpg. Consulta: 4 de marzo de 2015.

2. ELEMENTOS DEL CONTROL

El control de un sistema domótico se encargará de realizar las tareas de una casa, tomando las decisiones, en algunos casos, de lo que debe realizar, dependiendo de sensores o de las acciones que el operador del sistema quiera realizar en la casa, mediante una señal remota. De cualquiera de las dos formas el control es el cerebro del sistema, porque este va a decidir cómo realizar dicha acción.

En los elementos de un control domótico se encuentran los sensores que funcionan como los sentidos del sistema; los mismos le indican al sistema qué decisión tomar, dependiendo de un programa guardado en la memoria del sistema.

El cerebro del sistema o una unidad de procesamiento del control, es el encargado de tomar las decisiones, dependiendo de un programa de usuario, o también de los datos que le envían los sensores del sistema. Esta unidad de procesamiento controla las funciones de la casa, activando o desactivando los actuadores de la misma.

Los actuadores son los elementos finales del sistema que van a estar encargados de realizar las tareas de la casa, dependiendo de las decisiones del sistema. Las interfaces humano-máquina son los elementos desde donde el usuario del sistema controla qué funciones o parámetros debe tener el sistema, y la forma en la cual se va a comunicar con el control.

2.1. El microcontrolador PIC16F877A

Es un poderoso microcontrolador de 8 bit, de tecnología CMOS con memoria tipo Flash, que puede ejecutar instrucciones de hasta 200 nanosegundos de velocidad, es fácil de programar porque maneja sólo 35 instrucciones. Es fabricado por la empresa Microchip. Se puede encontrar en encapsulados de 40 a 44 pines. Cuenta con 8K palabras de 14 bits para la Memoria de Programa, tipo FLASH. 368 *bytes* de memoria de Datos RAM y 256 *bytes* de memoria de Datos EEPROM. Su consumo de potencia es muy bajo y además es completamente estático, esto quiere decir que el reloj puede detenerse y los datos de la memoria no se pierden.

2.1.1. Arquitectura del microcontrolador

En su interior está constituido por 3 unidades funcionales de una computadora, las cuales son: unidad central de procesamiento, memoria y periféricos de entrada y salida.

Un microcontrolador es un sistema completo con prestaciones limitadas que no pueden modificarse y que puede llevar a cabo las tareas para las que ha sido programado de forma autónoma.

2.1.1.1. Arquitectura interna de microcontrolador

Este término se refiere a los bloques funcionales internos que conforman el microcontrolador y la forma en que están conectados; por ejemplo: la memoria *FLASH* (de programa), la memoria RAM (de datos), los puertos, la lógica de control que permite que todo el conjunto funcione.

El PIC16F877A se basa en la arquitectura Harvard, la cual consiste en un esquema en el que el CPU está conectado a dos memorias por medio de dos buses separados. Una de las memorias contiene las instrucciones del programa, y es llamada memoria de programa. La otra memoria solo almacena los datos y es llamada memoria de datos. Ambos buses son totalmente independientes y pueden ser de distintos anchos. Utilizan un procesador de tecnología RISC.

El set de instrucciones y el bus de la memoria de programa pueden diseñarse de manera tal que todas las instrucciones tengan una sola posición de memoria de programa. Además, como los buses son independientes, el CPU puede estar accediendo a los datos para completar la ejecución de una instrucción, y al mismo tiempo estar leyendo la próxima instrucción a ejecutar. Esta misma estructura es la que permite la superposición de los ciclos de búsqueda y ejecución de las instrucciones, lo cual se ve reflejado en una mayor velocidad del microcontrolador.

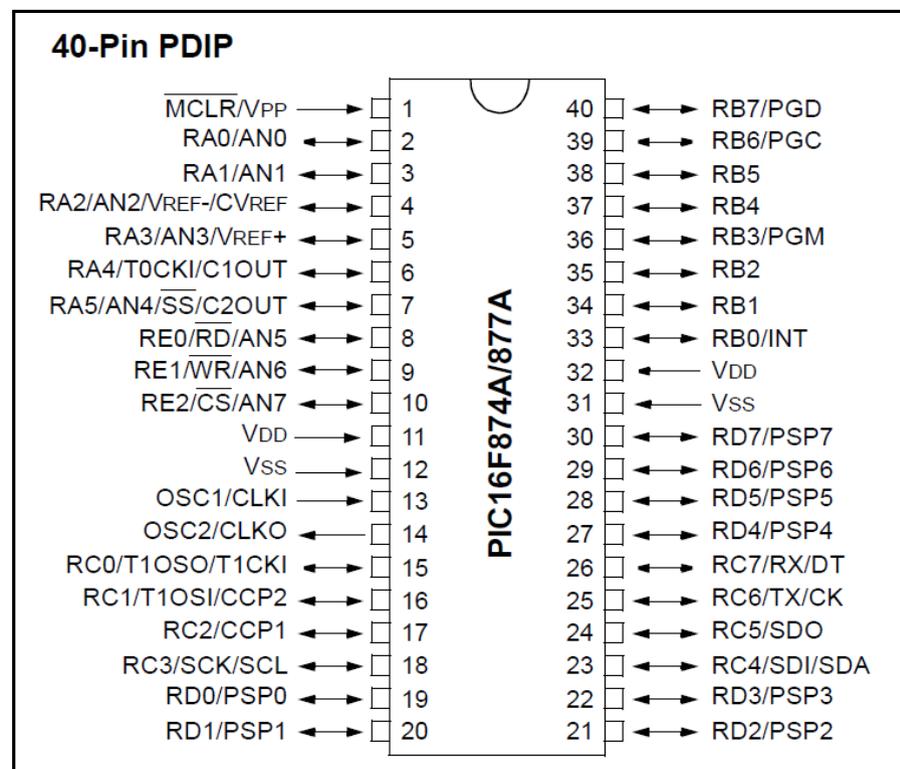
Todos los elementos se conectan entre sí por medio de buses. Un bus es un conjunto de líneas que transportan información entre dos o más módulos. Vale la pena destacar que el PIC16F877A tiene un bloque especial de memoria de datos de 256 bytes del tipo EEPROM, además de los dos bloques de memoria principales que son el de programa y el de datos o registros.

La figura 14 muestra la arquitectura general del PIC16F877A; en ella se pueden apreciar los diferentes bloques que lo componen y la forma en que se conectan. Se muestra la conexión de los puertos, las memorias de datos y de programa, los bloques especiales como el *watchdog*, los temporizadores de arranque, el oscilador, entre otros.

Los pines del puerto A y del puerto E pueden trabajar como entradas para el convertidor análogo a digital interno, es decir, allí se podría conectar una señal proveniente de un sensor o de un circuito analógico para que el microcontrolador la convierta en su equivalente digital. Los puertos B, C y D solo pueden trabajar con señales digitales.

La máxima capacidad de corriente de cada uno de los pines de los puertos en modo sumidero (*sink*) o en modo fuente (*source*) es de 25 mA .

Figura 15. **Distribución de los puertos del PIC16F877A**



Fuente: Microchip Technology Incorporated, *PIC16F87XA, Data Sheet*. p. 3.

Los puertos y sus terminales pueden clasificarse entre analógicos y digitales. A las terminales de entrada analógica se les reconoce por su nombre;

estas siempre empiezan con las letras AN seguidas del número de bit correspondiente a su puerto, por ejemplo, la terminal AN5 es una entrada analógica que corresponde al bit 5 de su puerto. Por otro lado, las terminales de entrada o salida digital se reconocen por la letra R seguida de la letra del puerto y el número de bits correspondientes a ese puerto. Por ejemplo, la terminal nombrada RD3 es una entrada o salida digital que corresponde al bit 3 del puerto D o PORTD.

2.1.2. Memoria Interna

La unidad de memoria es una parte del microcontrolador utilizada para almacenar los datos. El microcontrolador PIC16F877A tiene cuatro tipos de memoria: ROM, RAM, EEPROM y FLASH, teniendo cada una, funciones, características y organización específica.

2.1.2.1. Memoria de datos

La memoria de datos es el espacio que sirve para guardar datos que varían continuamente en el programa y en muchos casos se deben leer y escribir, por lo que para esta tarea el microcontrolador tiene distintos tipos de memoria, y estos pueden ser memoria tipo RAM y EEPROM. Cada tipo de memoria tiene distintas funciones.

2.1.2.1.1. Memoria RAM

Es la memoria de tipo volátil, la cual pierde el valor de sus datos cuando la alimentación del microcontrolador se pierde. Tiene dos áreas, las cuales cumplen con un trabajo específico.

- Registros de propósito general: se utilizan para almacenar los datos temporales y los resultados creados durante el funcionamiento. Por ejemplo, si el programa realiza un conteo (de los productos en una cadena de montaje), es necesario tener un registro que representa el resultado de la “suma”. Como el microcontrolador no es nada creativo, es necesario especificar la dirección de un registro de propósito general y asignarle esa función. Se debe crear un programa simple para incrementar el valor de este registro por 1, después de que cada producto haya pasado por el sensor. Ahora el microcontrolador puede ejecutar el programa ya que sabe qué es y dónde está la suma que se va a incrementar. De manera similar, a cada variable de programa se le debe preasignar alguno de los registros de propósito general.
- Registros de función especial: son parte de la memoria RAM cuyo propósito es predeterminado durante el proceso de fabricación y no se puede cambiar. Como los bits están conectados a los circuitos particulares en el chip (convertidor A/D y el módulo de comunicación serial), cualquier cambio de su contenido afecta directamente al funcionamiento del microcontrolador o de alguno de los circuitos. Por ejemplo, el registro ADCON0 controla el funcionamiento del convertidor A/D. Al cambiar los bits se determina qué pin del puerto se configurará como la entrada del convertidor, el momento del inicio de la conversión, así como la velocidad de la conversión.

Otra característica de estas localidades de memoria es que tienen nombres, lo que simplifica considerablemente el proceso de escribir un programa. Como el lenguaje de programación de alto nivel puede utilizar la lista de todos los registros con sus direcciones exactas, basta con especificar el nombre de registro para leer o cambiar su contenido.

El PIC16F877A posee cuatro bancos de memoria RAM; cada banco posee 128 *bytes*. De estos 128 los primeros 32 (hasta el 1Fh) son registros que cumplen un propósito especial en el control del microcontrolador y en su configuración. Los 96 siguientes son registros de propósito general.

2.1.2.1.2. Memoria EEPROM

Similar a la memoria de programa, el contenido de memoria EEPROM está permanentemente guardado al apagar la fuente de alimentación. Sin embargo, a diferencia de la ROM, el contenido de la EEPROM se puede cambiar durante el funcionamiento del microcontrolador. Es la razón por la que esta memoria es perfecta para guardar permanentemente algunos resultados creados y utilizados durante la ejecución del programa.

El tamaño de la EEPROM para un chip PIC16f877A es de 256 *bytes*. Hay que tener en cuenta que el tamaño de la memoria EEPROM interna del microcontrolador es pequeño, pero permite añadir módulos de memoria EEPROM externa de mayor tamaño.

2.1.2.2. Memoria de programa

Es una memoria del tipo FLASH; se puede programar y borrar eléctricamente, lo que facilita el desarrollo de los programas y la experimentación. En ella se graba o almacena el programa o códigos que el microcontrolador debe ejecutar. Esta memoria es no volátil; si el microcontrolador deja de ser alimentado eléctricamente, los datos que haya en esta memoria permanecerán.

El tamaño de la memoria Flash puede variar dependiendo del microcontrolador, aunque no es muy grande. Por ejemplo, para el chip PIC16F877A el tamaño de la memoria Flash es de 8 K de longitud con datos de 14 bits en cada posición.

2.1.3. Características básicas del microcontrolador PIC16F877A

- Memoria de programa: Flash, 8 K de instrucciones de 14 bits c/u.
- Memoria de datos: 368 *bytes* RAM, 256 *bytes* EEPROM.
- Pila (*stack*): 8 niveles (14 bits).
- Fuentes de interrupción: 13
- Instrucciones: 35
- Encapsulado: DIP de 40 pines.
- Frecuencia oscilador: 20 MHz (máxima)
- Temporizadores/Contadores: 1 de 8 bits (*timer 0*); 1 de 16 bits (*timer 1*); 1 de 8bits (*timer 2*) con pre y post escalador. Un perro guardián (WDT).
- Líneas de E/S : 6 del puerto A, 8 del puerto B, 8 del puerto C, 8 del puerto D y 3 del puerto E, además de 8 entradas análogas.
- Dos módulos de captura, comparación y PWM: captura: 16 bits. Resolución máx. = 12.5 nseg. Comparación: 16 bits. Resolución máx. = 200 nseg. PWM: Resolución máx. = 10 bits.
- Convertidor análogo/digital de 10 bits multicanal (8 canales de entrada).
- Puerto serial síncrono (SSP) con bus SPI (modo maestro) y bus I²C (maestro/esclavo).
- USART (*Universal synchronous asynchronous receiver transmitter*) con dirección de detección de 9 bits.
- Corriente máxima absorbida/suministrada (*sink/source*) por línea (pin): 25 mA.

- Oscilador: soporta 4 configuraciones diferentes: XT, RC, HS, LP.
- Tecnología de fabricación: CMOS.
- Voltaje de alimentación: 3.0 a 5.5 V DC.
- Puede operar en modo microprocesador.

2.1.4. Alimentación del microcontrolador

El voltaje de alimentación del microcontrolador es de 5V. Puede funcionar en el rango de los 3,0 a 5,5 voltios. A primera vista basta con encender una fuente de alimentación para hacer funcionar un microcontrolador, y basta con apagar una fuente de alimentación para detenerlo. En realidad, el arranque y el final del funcionamiento son las fases críticas de las que se encarga una señal especial denominada *reset*.

El microcontrolador tiene dos terminales marcados con V_{DD} , que es la fuente de alimentación positiva (+) y dos terminales marcados con V_{SS} que son la referencia de voltaje (GND); es importante que se conecten los cuatro terminales para que funcione de forma correcta.

2.1.5. El oscilador externo

Todo microcontrolador requiere un circuito externo que le indique la velocidad a la que debe trabajar. Este circuito que se conoce como oscilador o reloj es de vital importancia para el buen funcionamiento del sistema. EIPIC16F877A puede utilizar cuatro tipos de oscilador diferentes. Estos tipos son:

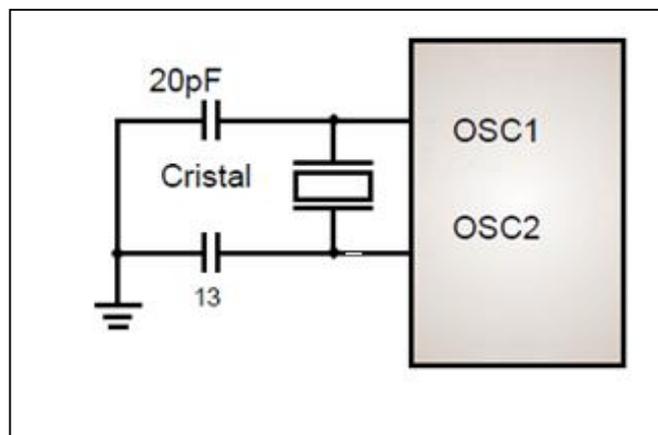
- RC. Oscilador con resistencia y condensador
- XT. Cristal (por ejemplo de 1 a 4 MHz)

- HS. Cristal de alta frecuencia (por ejemplo 10 a 20 MHz).
- LP. Cristal para baja frecuencia y bajo consumo de potencia.

En el momento de programar el microcontrolador se debe especificar qué tipo de oscilador se usa. Esto se hace a través de unos registros de configuración. Utiliza dos terminales para la conexión de un oscilador externo que están marcados como OSC1 y OSC2.

Si se conecta un tipo de oscilador XT con un cristal de 4 MHz, que garantiza precisión y es muy comercial. Internamente esta frecuencia es dividida por cuatro, lo que hace que la frecuencia efectiva de trabajo sea de 1 MHz en este caso, por lo que cada instrucción se ejecuta en un microsegundo. El cristal debe ir acompañado de dos condensadores y se conecta como se muestra en la figura 16.

Figura 16. **Conexión de un oscilador XT**



Fuente: elaboración propia, empleando Eagle 6.2.0.

2.1.6. **Reset**

En los microcontroladores se requiere un pin de *reset* para reiniciar el funcionamiento del sistema cuando sea necesario, ya sea por una falla que se presente o porque así se haya diseñado el sistema. El pin de reset en los PIC es llamado MCLR (*master clear*). Existen varias formas de reiniciar el sistema:

- Al encendido (*power on reset*).
- Pulso en el pin MCLR durante operación normal.
- Pulso en el pin MCLR durante el modo de bajo consumo (modo *sleep*).
- El rebase del conteo del circuito de vigilancia (*watchdog*) durante operación normal.
- El rebase del conteo del circuito de vigilancia (*watchdog*) durante el modo de bajo consumo (*sleep*).

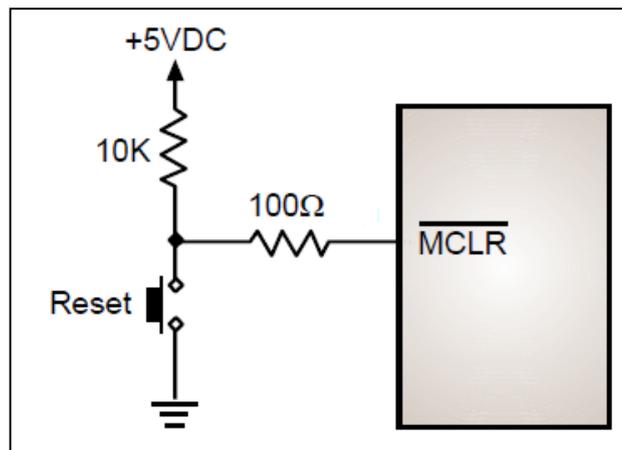
2.1.6.1. **El encendido del reset**

Se consigue gracias a dos temporizadores. El primero de ellos es el OST (*oscillator start-up timer*: temporizador de encendido del oscilador), orientado a mantener el microcontrolador en reset hasta que el oscilador del cristal es estable. El segundo es el PWRT (*Power-Up Timer*: Temporizador de encendido), que provee un retardo fijo de 72 ms (nominal) en el encendido únicamente, diseñado para mantener el dispositivo en reset mientras la fuente se estabiliza. Para utilizar estos temporizadores, solo basta con conectar el pin MCLR a la fuente de alimentación, evitándose utilizar las tradicionales redes de resistencias externas en el pin de *reset*.

El *reset* por MCLR se consigue llevando momentáneamente este pin a un estado lógico bajo, mientras que el *watchdog* WDT produce el *reset* cuando su

temporizador rebasa la cuenta, o sea que pasa de 0FFh a 00h. Cuando se quiere tener control sobre el *reset* del sistema se puede conectar un botón, como se muestra en la figura 17.

Figura 17. **Conexión del botón de *reset***



Fuente: elaboración propia, empleando Eagle 6.2.0.

2.1.7. Temporizadores

Una de las labores más habituales en los programas de control de dispositivos suele ser determinar intervalos concretos de tiempo, y recibe el nombre de temporizador (*timer*) el elemento encargado de realizar dicha función. También suele ser frecuente contar los impulsos que se producen en el exterior del sistema, y el elemento destinado para este fin se denomina contador. Los PIC16f877A poseen un temporizador/contador de 8 bits, llamado TMR0, que actúa de dos maneras diferentes:

- Como contador de sucesos que están representados por los impulsos que se aplican a la terminal RA4/TOCK1. Al llegar al valor FFh se desborda el

contador y, con el siguiente impulso pasa a 00 h, advirtiendo esta circunstancia, activando un señalizador o provocando una interrupción.

- Como temporizador, cuando se carga en el registro que implementa al recurso un valor inicial, se incrementa con cada ciclo de instrucción ($Fosc/4$) hasta que se desborda, o sea pasa de FF h a 00 h y avisa poniendo a 1 un bit señalizador o provocando una interrupción.

El microcontrolador PIC16f877A dispone de dos temporizadores: el TMR0 y el perro guardián (*Watchdog*). El primero actúa como principal y sobre él recae el control de tiempos y el conteo de impulsos. El otro vigila que el programa no se detenga por algún tipo de bloqueo, y para cada cierto tiempo comprueba si el programa se está ejecutando normalmente.

2.1.8. Interrupciones

Las interrupciones son desviaciones del flujo de control del programa principal, y son originadas asincrónicamente por diversos sucesos que no se hallan bajo la supervisión de las instrucciones. Dichos sucesos pueden ser externos al sistema, como la generación de un flanco o nivel activo en una terminal del microcontrolador, o bien internos, como el desbordamiento de un contador.

Se le llama interrupción a un evento especial que hace que el microcontrolador deje de realizar el programa principal, y se dedique a un subprograma especial llamado rutina de servicio de Interrupción. Este subprograma se ejecuta solamente cuando la interrupción tiene efecto, luego de lo cual el microcontrolador regresa a la instrucción donde se quedó del

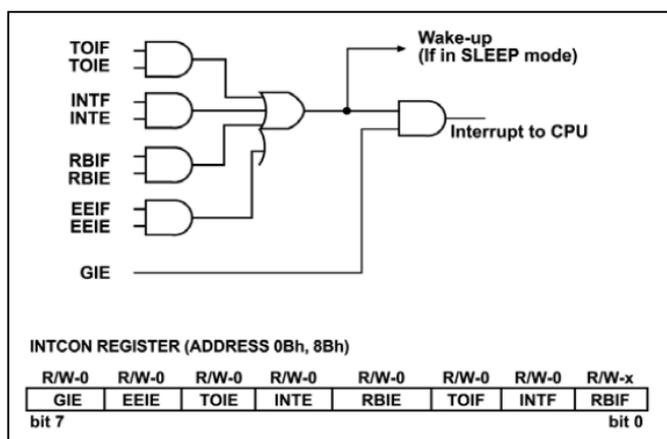
programa principal. Existen 4 fuentes de interrupción para el PIC16F877A que son:

- Interrupción externa en el pin RB0/INT
- Desbordamiento del módulo temporizador TIMER 0
- Interrupción por cambio de estado de los pines RB4 a RB7
- Interrupción por la periferia

Estas interrupciones se deben activar o desactivar desde el programa a través del registro defunción especial llamado INTCON (control de interrupciones).

En la figura 18 se observa la lógica de funcionamiento de las interrupciones y el registro INTCON.

Figura 18. **Lógica de funcionamiento de las interrupciones y registro INTCON**



Fuente: Microchip Technology Incorporated. *PIC16F87XA, Data Sheet*, p. 55.

2.1.9. Puerto serial

El módulo de comunicación serial USART (*Universal synchronous asynchronous receiver transmitter*) disponible en los microcontroladores PIC16F877A, es un poderoso sistema capaz de comunicarse con el protocolo RS232 hacia computadores y terminales que obedezcan el mismo protocolo.

El USART es uno de los primeros sistemas de comunicación serie. Las versiones nuevas de este sistema están actualizadas y se les denomina un poco diferente: EUSART.

El módulo transmisor/receptor universal síncrono/asíncrono mejorado (*Enhanced universal synchronous asynchronous receiver transmitter - EUSART*) es un periférico de comunicación serie de entrada/salida. Asimismo es conocido como interfaz de comunicación serie (*Serial communications Interface - SCI*). Contiene todos los generadores de señales de reloj, registros de desplazamiento y *bufers* de datos necesarios para realizar transmisión de datos serie de entrada/salida, independientemente de la ejecución de programa del dispositivo.

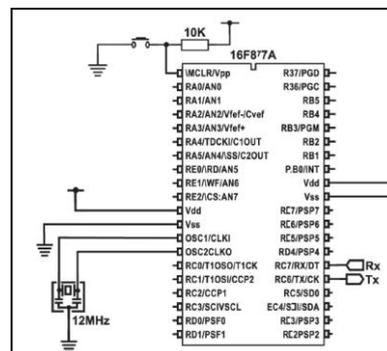
Como indica su nombre, aparte de utilizar el reloj para la sincronización, este módulo puede establecer la conexión asíncrona, lo que lo hace único para algunas aplicaciones. Por ejemplo, en caso de que sea difícil o imposible proporcionar canales especiales para transmisión y recepción de datos y señales de reloj (por ejemplo, mando a distancia de radio o infrarrojas), el módulo EUSART es definitivamente la mejor opción posible.

El USART integrado en el PIC16F877A posee las siguientes características:

- Transmisión y recepción asíncrona en modo *Full-duplex*
- Caracteres de anchura de 8 – 9 bits programables
- Detección de dirección en modo de 9 bits
- Detección de errores por saturación del *buffer* de entrada
- Comunicación *Half Duplex* en modo síncrono

Para utilizar el módulo USART, primero se debe definir la terminal RC7/RX/DT como entrada digital y luego la terminal RC6/TX/CK como salida digital. Después debe establecer la frecuencia de comunicación en baudios que puede estar entre los siguientes valores: 300, 600, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 31250, 38400 y 57600. Es necesario aclarar que entre mayor sea la frecuencia de comunicación habrá más probabilidad de errores de transferencia de datos. Una frecuencia normal de comunicación es de 9600 baudios, que ofrece rapidez y confiabilidad.

Figura 19. **Conexión del microcontrolador 16F877A para hacer una comunicación serial**



Fuente: elaboración propia, empleando Eagle 6.2.0.

2.1.10. Programación del microcontrolador

Se necesita programar el microcontrolador para que sea capaz de ejecutar un programa. Esto se denomina el código ejecutable y está compuesto por una serie de ceros y unos, aparentemente sin significado. Dependiendo de la arquitectura del microcontrolador, el código binario está compuesto por palabras de 12, 14 o 16 bits de anchura. Cada palabra es interpretada por la CPU como una instrucción a ser ejecutada durante el funcionamiento del microcontrolador.

Como es más fácil trabajar con el sistema de numeración hexadecimal, el código ejecutable se representa con frecuencia como una serie de los números hexadecimales denominado código Hex. A todas las instrucciones que el microcontrolador puede reconocer y ejecutar se le denominan colectivamente juego de instrucciones. En los microcontroladores PIC con las palabras de programa de 14 bits de anchura, el conjunto de instrucciones tiene 35 instrucciones diferentes.

Normalmente los programas se escriben en el lenguaje ensamblador cuando se requiere controlar completamente la ejecución de programa. Como el proceso de escribir un código ejecutable era considerablemente arduo, en consecuencia fue creado el primer lenguaje de programación denominado ensamblador (ASM). El proceso de programación se hizo un poco más complicado. Por otro lado, el proceso de escribir un programa dejó de ser un problema; las instrucciones en ensamblador consisten en las abreviaturas con significado.

Un programa denominado ensamblador instalado en la PC compila las instrucciones del lenguaje ensamblador a código máquina (código binario). Este programa compila instrucción a instrucción sin optimización. La ventaja principal

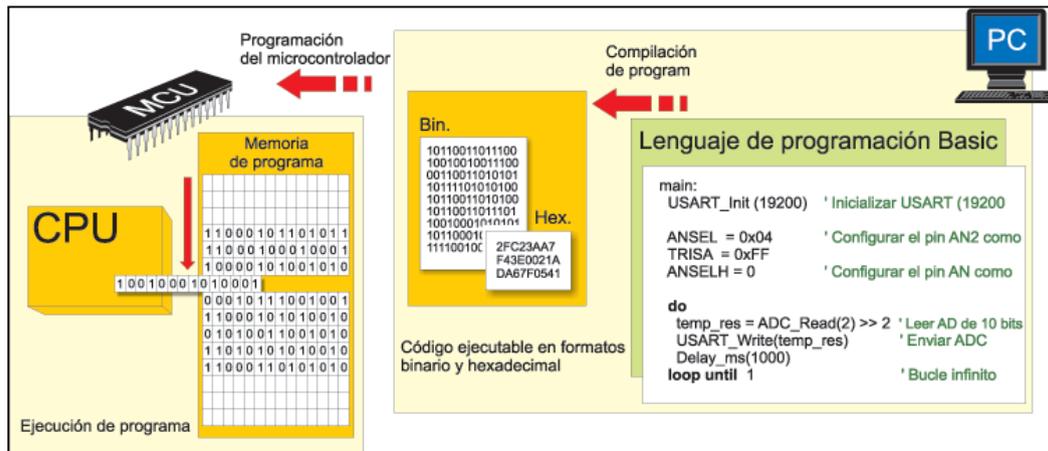
del lenguaje ensamblador es su simplicidad y el hecho de que a cada instrucción le corresponde una localidad de memoria. Como permite controlar todos los procesos puestos en marcha dentro del microcontrolador, este lenguaje de programación todavía sigue siendo popular.

Por otro lado, los programas se ejecutan siempre a alta velocidad y en la mayoría de casos no es necesario saber en detalle qué ocurre dentro del microcontrolador. A pesar de todos los lados buenos del lenguaje ensamblador, los programadores siempre han necesitado un lenguaje de programación similar al lenguaje utilizado en el habla cotidiana. Por último, los lenguajes de programación de alto nivel (Basic, entre otros) fueron creados. La ventaja principal de estos lenguajes es la simplicidad de escribir un programa. Varias instrucciones en ensamblador se sustituyen por una sentencia en Basic.

El programador ya no tiene que conocer el conjunto de instrucciones del microcontrolador utilizado. Ya no es posible conocer exactamente cómo se ejecuta cada sentencia, de todas formas ya no importa. Aunque siempre se puede insertar en el programa una secuencia escrita en ensamblador.

El proceso de escribir un programa en Basic: al seleccionar la opción apropiada, el programa será compilado en el ensamblador y luego en el código hex que será cargado en el microcontrolador. Similar al lenguaje ensamblador, un programa especializado e instalado en la PC se encarga de compilar un programa a código máquina. A diferencia del ensamblador, los compiladores para los lenguajes de programación de alto nivel crean un código ejecutable que no es siempre tan corto como el código escrito en ensamblador.

Figura 20. **Ejemplo de un programa en lenguaje Basic, la compilación y la programación de la memoria de programa**



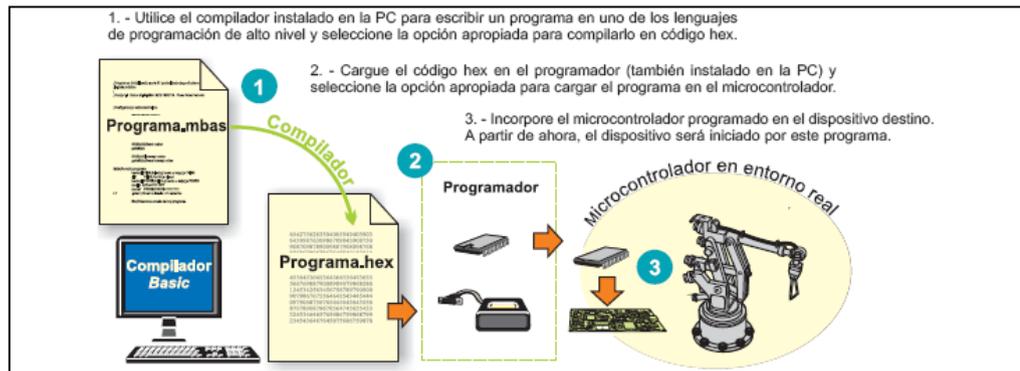
Fuente: *Mikroelectronica*. <http://www.mikroe.com/chapters/view/85/libro-de-la-programacion-de-los-microcontroladores-en-basic-capitulo-2-programacion-de-los-microcontroladores/>. Consulta: 10 de abril de 2015.

2.1.10.1. Lenguaje de programación *Basic*

Similar al uso de cualquier lengua que no está limitada a los libros y a las revistas, el lenguaje de programación Basic no está estrechamente relacionado con un tipo particular de ordenador, procesador o sistema operativo. Esto puede ser un problema, ya que Basic varía ligeramente dependiendo de su aplicación (como diferentes dialectos de una lengua).

El Basic es un lenguaje de programación simple y fácil de entender. Para utilizarlo correctamente, basta con conocer solo unos pocos elementos básicos en los que consiste cada programa. Estos son: identificadores, comentarios, operadores, expresiones, instrucciones, constantes, variables, símbolos, directivas, etiquetas, módulos, procedimientos y funciones.

Figura 21. **El proceso de la compilación de programa escrito en Basic en código Hex**



Fuente: *Mikroelectronica*. <http://www.mikroe.com/chapters/view/85/libro-de-la-programacion-de-los-microcontroladores-en-basic-capitulo-2-progamacion-de-los-microcontroladores/>. Consulta: 10 de abril de 2015.

2.2. Sensores

Dispositivo formado por células sensibles que detecta variaciones en una magnitud física y las convierte en señales normalmente eléctricas, útiles para un sistema de medida o control.

Los términos sensor y transductor se suelen aceptar como sinónimos, aunque, si hubiera que hacer alguna distinción, el término transductor es quizás más amplio, incluyendo una parte sensible o captador propiamente dicho y algún tipo de circuito de acondicionamiento de la señal detectada. Al centrarse en el estudio de los transductores cuya salida es una señal eléctrica, se puede dar la siguiente definición: “un transductor es un dispositivo capaz de convertir el valor de una magnitud física en una señal eléctrica codificada, va sea en forma analógica o digital”.

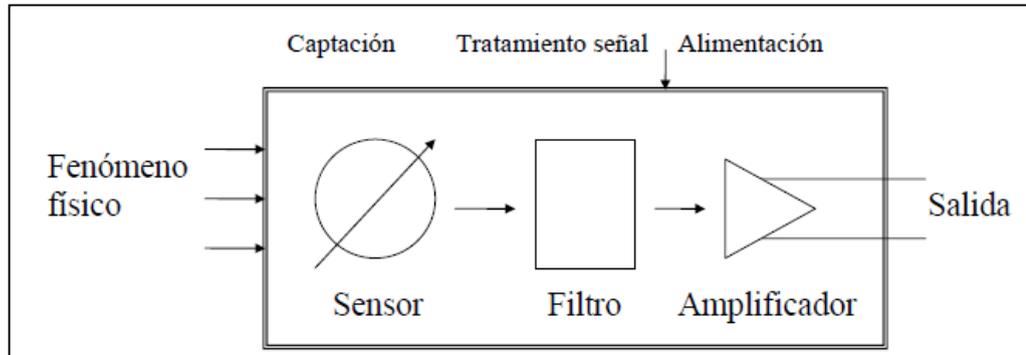
No todos los transductores tienen por qué dar una salida en forma de señal eléctrica. Como ejemplo puede valer el caso de un termómetro basado en la diferencia de dilatación de una lámina bimetálica, donde la temperatura se convierte directamente en un desplazamiento de una aguja indicadora.

Sin embargo, el término transductor suele asociarse bastante a dispositivos cuya salida es alguna magnitud eléctrica o magnética.

Los transductores basados en fenómenos eléctricos o magnéticos suelen tener una estructura general como la que muestra la figura 22, en la cual se distinguen las siguientes partes:

- Elemento sensor o captador: convierte las variaciones de una magnitud física en variaciones de una magnitud eléctrica o magnética, lo que se conoce como señal.
- Bloque de tratamiento de señal: si existe, suele filtrar, amplificar, linealizar y, en general, modificar la señal obtenida en el captador, por regla general utilizando circuitos electrónicos.
- Etapa de salida: esta etapa comprende los amplificadores, interruptores, conversores de código, transmisores y en general, todas aquellas partes que adaptan la señal a las necesidades de la carga exterior.

Figura 22. Estructura genérica de un transductor



Fuente: *Slideshare*. <http://es.slideshare.net/bryandree/sensores-y-actuadores-16167568>.
Consulta: 10 de abril de 2015.

2.2.1. Clasificación de sensores según su tipo de señal de salida

Atendiendo a la forma de codificar la magnitud medida se puede establecer una clasificación en:

- Analógicos: aquellos que dan como salida un valor de tensión o corriente variable en forma continua dentro del campo de medida. Es frecuente para este tipo de transductores que incluyan una etapa de salida para suministrar señales normalizadas de 0-10 V o 4-20 mA.
- Digitales: son aquellos que dan como salida una señal codificada en forma de pulsos o en forma de una palabra digital codificada en binario. BCD u otro sistema cualquiera.
- Todo-nada: Indican únicamente cuándo la variable detectada rebasa un cierto umbral o límite. Pueden considerarse como un caso límite de los sensores digitales en el que se codifican solo dos estados.

2.2.2. Sensores pasivos y activos

Otro criterio de clasificación, relacionado con la señal de salida es el hecho de que el captador propiamente dicho requiera o no una alimentación externa para su funcionamiento.

- Los sensores pasivos se basan, por lo general, en la modificación de la impedancia eléctrica o magnética de un material bajo determinadas condiciones físicas o químicas (resistencia, capacidad, inductancia, reluctancia, entre otros.). Este tipo de sensores, debidamente alimentados, provoca cambios de tensión o de corriente en un circuito, los cuales son recogidos por el circuito de interfaz.
- Los sensores activos son en realidad generadores eléctricos, generalmente de pequeña señal. Por ello no necesitan alimentación exterior para funcionar, aunque si suelen necesitarla para amplificar la débil señal del captador.

2.2.3. Características generales de los sensores

La relación salida/entrada en régimen permanente depende casi exclusivamente del bucle de realimentación. Así pues, dejando a un lado las características constructivas particulares de cada transductor o de cada sistema de medida previsto como lazo de realimentación, es importante conocer diversos aspectos genéricos de su comportamiento a fin de prever o corregir la actuación tanto estática como dinámica del lazo de control.

2.2.3.1. Características estáticas

Describen la actuación del sensor en régimen permanente o con cambios muy lentos de la variable a medir.

- El campo de medida es el rango de valores de la magnitud de entrada comprendido entre el máximo y el mínimo detectables por un sensor, con una tolerancia de error aceptable.
- Resolución: indica la capacidad del sensor para discernir entre valores muy próximos de la variable de entrada. Se mide por la mínima diferencia entre dos valores próximos que el sensor es capaz de distinguir. Se puede indicar en términos de valor absoluto de la variable física medida o en porcentaje respecto del fondo de escala de la salida.
- Precisión: define la máxima desviación entre la salida real obtenida de un sensor en determinadas condiciones de entorno y el valor teórico de dicha salida que correspondería, en idénticas condiciones, según el modelo ideal especificado como patrón. Se suele indicar en valor absoluto de la variable de entrada o en porcentaje sobre el fondo de escala de la salida.
- Repetibilidad: característica que indica la máxima desviación entre valores de salida obtenidos al medir varias veces un mismo valor de entrada, con el mismo sensor y en idénticas condiciones ambientales. Se suele expresar en porcentaje referido al fondo de escala y da una indicación del error aleatorio del sensor. Algunas veces se suministran datos de repetibilidad, variando ciertas condiciones ambientales, lo cual permite obtener las derivas ante dichos cambios.

- Linealidad: se dice que un transductor es lineal si existe una constante de proporcionalidad única que relaciona los incrementos de señal de salida con los correspondientes incrementos de señal de entrada, en todo el campo de medida. La no linealidad se mide por la máxima desviación entre la respuesta real y la característica puramente lineal, referida al fondo de escala.
- Sensibilidad: característica que indica la mayor o menor variación de la salida por unidad de la magnitud de entrada. Un sensor es tanto más sensible cuanto mayor sea la variación de la salida producida por una determinada variación de entrada.
- Ruido: es cualquier sistema de medida que produce una desviación de la salida respecto del valor teórico.
- Histéresis: se dice que un transductor presenta histéresis cuando, a igualdad de la magnitud de entrada, la salida depende de si dicha entrada se alcanzó con aumentos en sentido creciente o en sentido decreciente. Se suele medir en términos de valor absoluto de la variable física o en porcentaje sobre el fondo de escala. Obsérvese que la histéresis puede no ser constante en todo el campo de medida.

2.2.3.2. Características dinámicas

Son las que describen la actuación del sensor en régimen transitorio, a base de dar su respuesta temporal ante determinados estímulos estándar o a base de identificar el comportamiento del transductor con sistemas estándar.

2.2.3.2.1. Velocidad de respuesta

La velocidad de respuesta mide la capacidad de un transductor para que la señal de salida siga sin retraso las variaciones de la señal de entrada. La forma de cuantificar este parámetro es a base de una o más constantes de tiempo, que suelen obtenerse de la respuesta al escalón. Los parámetros más relevantes empleados en la definición de la velocidad de respuesta son los siguientes:

- Tiempo de retardo: tiempo transcurrido desde la aplicación del escalón de entrada hasta que la salida alcanza el 10 % de su valor permanente.
- Tiempo de subida: tiempo transcurrido desde que la salida alcanza el 10% de su valor permanente hasta que llega por primera vez al 90 % de dicho valor.
- Tiempo de establecimiento al 99 %: tiempo transcurrido desde la aplicación de un escalón de entrada hasta que la respuesta alcanza el régimen permanente, con una tolerancia de ± 1 .

2.2.3.2.2. Constante de tiempo

Para un transductor con respuesta de primer orden (una sola constante de tiempo dominante) se puede determinar la constante de tiempo a base de medir el tiempo empleado para que la salida alcance el 63 % de su valor de régimen permanente, cuando a la entrada se le aplica un cambio en escalón.

2.2.3.2.3. Respuesta frecuencial

Relación entre la sensibilidad y la frecuencia cuando la entrada es una excitación senoidal. Se suele indicar gráficamente mediante un gráfico de Bode. La respuesta frecuencial está muy directamente relacionada con la velocidad de respuesta.

2.2.3.2.4. Estabilidad y derivas

Características que indican la desviación de salida del sensor al variar ciertos parámetros exteriores distintos del que se pretende medir, tales como condiciones ambientales, alimentación u otras perturbaciones.

2.3. Actuadores

Dispositivo capaz de transformar energía eléctrica, neumática o hidráulica en la activación de un proceso físico. Este recibe la orden de un regulador o controlador y en función de ella genera la orden para activar un elemento final. La gama de posibles actuadores que puede analizar un control electrónico es enormemente extensa y variada. Entre los más habituales se encuentran los destinados a producir movimiento (motores y cilindros), los destinados a trasiego de fluidos (bombas) y los de tipo térmico (hornos, intercambiadores, entre otros.). Sin embargo, para una automatización de casa hay varios tipos de actuadores que interesan, tales como servomotores y servoválvulas.

2.3.1. Actuadores eléctricos

Es un dispositivo de movimiento accionado eléctricamente, puede ser un posicionador electrónico, un servomotor, una compuerta motorizada, entre

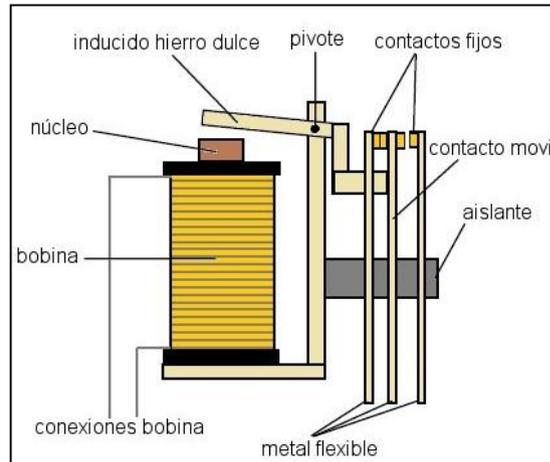
otros. La estructura de un actuador eléctrico es simple en comparación con la de los actuadores hidráulicos y neumáticos, ya que solo se requiere de energía eléctrica como fuente de poder.

Como se utilizan cables eléctricos para transmitir electricidad y la señal es altamente versátil y prácticamente no hay restricciones respecto de la distancia entre la fuente de poder y el actuador.

Existen preactuadores que se encargan de manejar la alta corriente o el accionamiento de los actuadores, estos son los que reciben la señal del control y hacen que funcione el actuador. Entre ellos están los siguientes:

- Relé: dispositivos electromagnéticos que conectan o desconectan un circuito eléctrico de potencia al excitar un electroimán o bobina de mando. Los relés se suelen emplear como etapa previa para accionar dispositivos más potentes como los propios contactores, electroválvulas u otros. El relé separa, en general, la parte de mando que trabaja con tensiones y corrientes débiles, de la parte de potencia, con tensiones y corrientes más elevadas. Muchas etapas de salida de controles utilizan relés cuya bobina va gobernada directamente por los circuitos electrónicos y que aportan la ventaja de aislar eléctricamente el circuito electrónico de los contactos de utilización. Su diagrama interno se muestra en la figura 23.

Figura 23. Partes de un relé



Fuente: *Electrónica fácil*. <http://st-elf.electronicafacil.net/tutoriales/147/rele1b.gif>. Consulta: 10 de abril 2015.

- **Contactador:** la diferencia entre relé y contactador está precisamente en la potencia que es capaz de seccionar cada uno. Los relés están previstos para accionar pequeñas potencias, generalmente inferiores a 1 kW. Mientras que los contactores pueden accionar grandes potencias (centenares de kilovatios). Los contactos principales son los destinados a las maniobras del circuito de potencia de los montajes tales como alimentación de motores eléctricos, calefactores, electrodomésticos, entre otros. El circuito electromagnético, la bobina y la espira realizan idéntica función a la del relé; la diferencia estriba en el tamaño y algún otro detalle. Los relés y los contactores son elementos clave para la realización de circuitos de mando y de fuerza, tanto en la implementación de automatismos eléctricos como en los basados en tecnologías programables, usados como elementos de pre accionamiento.

- Relés de estado sólido: los relés de estado sólido SSR. Están formados un circuito electrónico que contiene en su interior un circuito disparado por nivel, acoplado a un interruptor semiconductor, un transistor o un tiristor. Tienen la ventaja de no producir ruido debido a que el circuito de acoplamiento está basado en componentes no electromecánicos. Este relé es indicado para conmutar cargas resistivas o poco conductivas así como la conexión de lámparas de incandescencia, conexión de calefactores y señalización. Otra ventaja que muestran estos tipos de relés es que no producen chispa al momento de la conexión y eso es importante debido que pueden trabajar en ambientes explosivos.

2.3.1.1. Motor eléctrico

El motor eléctrico es sin duda el actuador eléctrico más utilizado, son máquinas eléctricas rotatorias, que transforman una energía eléctrica en energía mecánica. Tienen múltiples ventajas, entre las que cabe citar su economía, limpieza, comodidad y seguridad de funcionamiento, el motor eléctrico ha reemplazado en gran parte a otras fuentes de energía, tanto en la industria como en el transporte, las minas, el comercio, o el hogar. Existe una gama muy amplia de tipo de motores, que se elegirán en función de la aplicación y de la red eléctrica que se disponga en el lugar de utilización.

En los ámbitos de la domótica, los motores forman parte como accionamiento lineal o rotatorio de algún tipo de equipo con el que gobiernan las múltiples instalaciones, cubriendo determinadas funcionalidades: climatización, bombeo, apertura y cierre de portones, persianas, cortinas y toldos, sistemas de elevación, ventilación, entre otros, formando parte de los más diversos electrodomésticos y equipos electrónicos: lavadoras, licuadoras, bombas de acuario, cámaras motorizadas, así como otros.

Entre los más comúnmente utilizados están los motores de corriente continua y los de corriente alterna, motores paso a paso, los motores *brushless*.

2.3.1.1.1. Motor eléctrico de corriente alterna monofásico

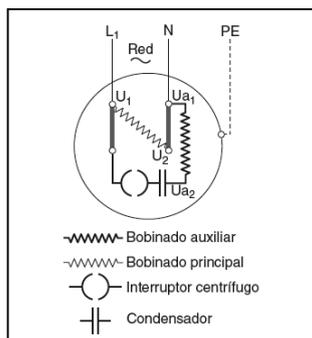
En el ámbito doméstico tienen gran aplicación los motores eléctricos, por lo que es necesario que estos puedan funcionar en redes monofásicas. Los motores monofásicos son muy parecidos a los trifásicos, con el inconveniente de que su rendimiento y factor de potencia son inferiores. A igual potencia, el monofásico es más voluminoso que el trifásico y, siempre que las condiciones lo permitan, se utilizarán trifásicos. Los más utilizados son:

- Motor monofásico con bobinado auxiliar de arranque: como todos los motores eléctricos, está formado por un circuito magnético y dos eléctricos. El circuito magnético está formado por el estator, donde se coloca el bobinado inductor y el rotor que incorpora el bobinado inducido, que en la mayoría de los casos suele ser de jaula de ardilla. De su nombre se desprende que utiliza un solo bobinado inductor, recorrido por una corriente alterna que crea un flujo también alterno, pero de dirección constante que, por sí solo, no es capaz de hacer girar al rotor. Si el rotor se encuentra ya girando, en los conductores del bobinado rotórico se generan fuerzas electromotrices que hacen que por el bobinado rotórico circulen corrientes, que a su vez generan un flujo de reacción desfasado 90 grados eléctricos respecto del principal.

La interacción entre estos dos flujos hace que el motor se comporte como un motor bifásico y el rotor continúe girando. De lo expresado

anteriormente se desprende que el motor monofásico es incapaz de arrancar por sí solo, pero si se pone en marcha, se mantiene funcionando de forma normal hasta su desconexión. Por ello, hay que dotar a dicho motor de un dispositivo adecuado para iniciar el arranque. El más utilizado es incorporar al estator un bobinado auxiliar que funciona durante el periodo de arranque y que se desconecta una vez que el motor está en funcionamiento.

Figura 24. **Motor monofásico con bobinado auxiliar y condensador**



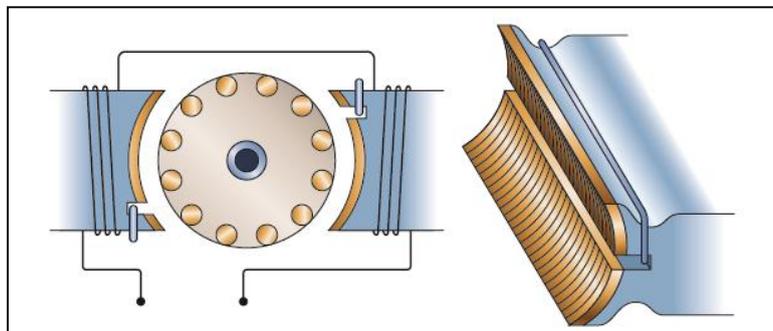
Fuente: *McGraw-Hill*. www.mcgraw-hill.es/bcv/guide/capitulo/8448173104.pdf. Consulta: 10 de abril 2015.

En estas condiciones, el motor en el arranque es un motor bifásico, con sus bobinados desfasados 90 grados eléctricos, que hace que el motor se ponga en marcha. Una vez alcanzado el régimen de vueltas, se desconecta el bobinado auxiliar, de forma que queda funcionando como motor monofásico. Para realizar la desconexión del bobinado auxiliar se utilizan los interruptores centrífugos acoplados en el eje del motor. Los bobinados se conectan en paralelo a la placa de bornes. El motor monofásico tiene un rendimiento, par de arranque y factor de potencia algo bajos. Para compensar dichos valores, se recurre a conectar en serie

con el bobinado auxiliar un condensador electrolítico, con lo que se consiguen valores de rendimiento y par de arranque mucho mejores.

- Motor de espira en cortocircuito: el motor de espira en cortocircuito está constituido por un estator de polos salientes y un rotor de jaula de ardilla. En la masa polar se incorpora una espira en cortocircuito que abarca un tercio aproximadamente del polo. Las bobinas rodean las masas polares, como se muestra en la figura 25.

Figura 25. **Esquema de motor monofásico de espira en cortocircuito**



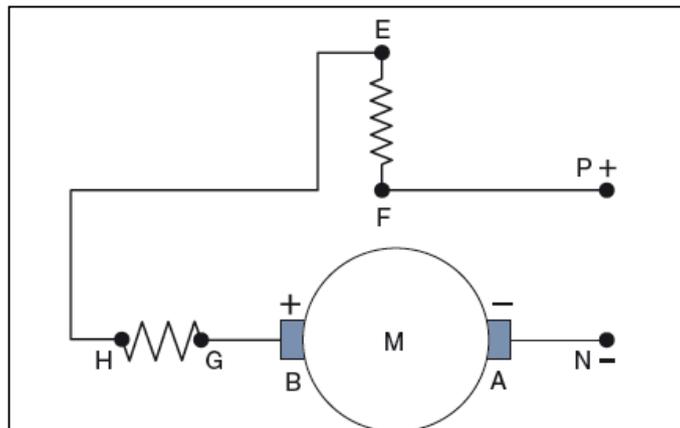
Fuente: *McGraw-Hill*. www.mcgraw-hill.es/bcv/guide/capitulo/8448173104.pdf. Consulta: 10 de abril 2015.

Al alimentar las bobinas polares con una corriente alterna se produce un campo magnético alterno en el polo que por sí solo no es capaz de poner en marcha el motor. El flujo que atraviesa la espira genera en esta una fuerza electromotriz inducida que hace que circule una corriente de elevado valor por la espira. Esto a su vez crea un flujo propio que se opone al flujo principal. En estas condiciones se obtiene un sistema de dos flujos en el que el flujo propio estará en retraso respecto del flujo principal, haciendo que el motor gire.

El sentido de giro será siempre el que va desde el eje del polo hacia la espira en cortocircuito colocada en el mismo. Si por algún motivo se necesitará invertir el giro, se tendría que desmontar el motor e invertir todo el conjunto del rotor manteniendo la posición del estátor. Dado que estos motores tienen un rendimiento muy bajo, su utilización se limita a pequeñas potencias de hasta 300 W y para trabajos de ventilación, bombas de desagües de electrodomésticos, entre otros.

- Motor universal: es un motor monofásico que puede funcionar tanto en corriente continua como alterna. Su constitución es esencialmente la del motor serie de corriente continua, y sus características de funcionamiento son análogas. En la figura 26 se presenta la forma esquemática de este motor.

Figura 26. **Esquema de conexiones del motor universal**



Fuente: *McGraw-Hill*. www.mcgraw-hill.es/bcv/guide/capitulo/8448173104.pdf. Consulta: 10 de abril 2015.

El motor serie de corriente continua se caracteriza por tener un fuerte par de arranque y su velocidad está en función inversa a la carga, llegando a

embalarse cuando funciona en vacío. Funcionando en corriente alterna, este inconveniente se ve reducido porque su aplicación suele ser en motores de pequeña potencia y las pérdidas por rozamientos, cojinetes, entre otros, son elevadas respecto del total, por lo que no presentan el peligro de embalarse, pero sí alcanzan velocidades de hasta 20 000 revoluciones por minuto (rpm), que los hace bastante idóneos para pequeños electrodomésticos y máquinas herramientas portátiles.

El motor universal es, sin duda, el más utilizado en la industria del electrodoméstico. Tienen la ventaja de poder regular la velocidad sin grandes inconvenientes.

Para que un motor de este tipo pueda funcionar con corriente alterna, es necesario que el empilado de su inductor (el núcleo de los electroimanes) sea de chapa magnética para evitar las pérdidas en el hierro. El bobinado inductor de los motores universales suele ser bipolar, con dos bobinas inductoras. El motor universal funciona en corriente continua exactamente igual que un motor serie. Si el motor se alimenta con corriente alterna, arranca por sí solo, ya que la corriente que recorre el bobinado inductor presenta cien alternancias por segundo; lo mismo le ocurre a la corriente que recorre el bobinado inducido, por lo que el momento de rotación y el sentido de giro permanecen constantes.

2.3.1.1.2. Motor eléctrico de corriente directa

Se utilizan en casos en los que es de importancia el poder regular continuamente la velocidad del eje y en aquellos casos en los que se necesita de un toque de arranque elevado. Además, utilizan cuando es imprescindible

utilizar corriente continua, como es el caso de trenes y automóviles eléctricos, motores para utilizar en el arranque y en los controles de automóviles, motores accionados a pilas o baterías, entre otros.

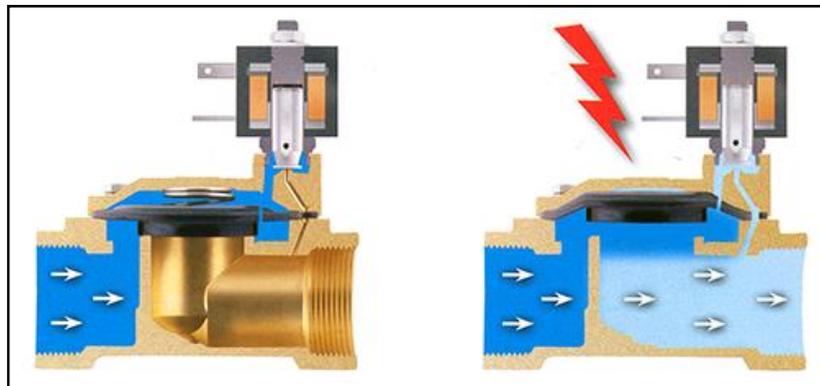
Para funcionar, el motor de corriente continúa o directa precisa de dos circuitos eléctricos distintos: el circuito de campo magnético y el circuito de la armadura.

El campo (básicamente un imán o un electroimán) permite la transformación de energía eléctrica recibida por la armadura en energía mecánica entregada a través del eje. La energía eléctrica que recibe el campo se consume totalmente en la resistencia externa con la cual se regula la corriente del campo magnético. Es decir, ninguna parte de la energía eléctrica recibida por el circuito del campo es transformada en energía mecánica. El campo magnético actúa como una especie de catalizador que permite la transformación de energía en la armadura.

La armadura consiste en un grupo de bobinados alojados en el rotor y en un ingenioso dispositivo denominado colector, mediante el cual se recibe corriente continua desde una fuente exterior y se convierte la correspondiente energía eléctrica en energía mecánica que se entrega a través del eje del motor. En la transformación se pierde un pequeño porcentaje de energía en los carbones del colector, en el cobre de los bobinados, en el hierro (por corriente parásita e histéresis), en los rodamientos del eje y la fricción del rotor por el aire.

Las electroválvulas se aplican a surtidores automáticos de combustibles, irrigación de parques, fuentes de agua danzantes, dosificadores de líquidos o gases, regulación de niveles de líquidos en máquinas envasadoras, lavaderos automáticos de autos, máquinas de limpieza, procesos de niquelado o galvanizado, en máquinas de café y en muchos lugares más.

Figura 28. **Electroválvula servopilotadas**



Fuente: *Suministros industriales PASAI*. <http://www.pasaisa.com/blog/electrovalvulas/>. Consulta: 10 de abril de 2015.

2.4. **Comunicación Bluetooth**

La comunicación inalámbrica supuso un gran salto tanto cuantitativo como cualitativo en la gestión de la información, permitiendo el acceso e intercambio de la misma de forma remota, sin necesidad de una conexión física vía cable. Tradicionalmente, la comunicación inalámbrica se realizaba vía radio operando en diversas bandas de frecuencias o bien a través de dispositivos de comunicación basados en infrarrojos.

La transmisión inalámbrica de voz y datos ha permanecido en constante evolución, surgiendo nuevos estándares entre los que destacan Bluetooth o el

desarrollado en la norma IEEE 802.11b. Bluetooth es una tecnología orientada a sistemas de comunicación a corta/media distancia y optimizados para un bajo coste y menor consumo, posicionándose como la tecnología del futuro para pequeñas redes o sistemas de captación de información.

Bluetooth es la norma que define un estándar global de comunicación inalámbrica que posibilita la transmisión de voz y datos entre diferentes equipos mediante un enlace por radiofrecuencia.

Los principales objetivos que se pretende conseguir con esta norma son:

- Facilitar las comunicaciones entre equipos móviles y fijos.
- Eliminar cables y conectores entre estos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre los equipos personales.

La especificación de Bluetooth define un canal de comunicación de máximo 720 Kb/s con un rango óptimo de 10 metros (opcionalmente 100 m).

La frecuencia de radio con la que trabaja está en el rango de 2.4 a 2.8 GHZ con amplio espectro y saltos de frecuencia con posibilidad de transmitir en *full dúplex* con un máximo de 1600 saltos/seg. Los saltos de frecuencia se dan entre un total de 79 frecuencias con intervalos de 1 MHz; esto permite dar seguridad y robustez.

El protocolo de banda base (canales simples por línea) combina *switching* de circuitos y paquetes. Para asegurar que los paquetes no lleguen fuera de orden, los slots pueden ser reservados por paquetes síncronos, un salto diferente de señal es usado para cada paquete. Por otro lado, el *switching* de

circuitos puede ser asíncrono o síncrono. Tres canales de datos síncronos (voz), o un canal de datos síncrono y otro asíncrono, pueden ser soportados en un solo canal; cada canal de voz puede soportar una tasa de transferencia de 64 Kbps en cada sentido, la cual es suficientemente adecuada para la transmisión de voz. Un canal asíncrono puede transmitir como mucho 721 Kbps en una dirección y 56 Kbps en la dirección opuesta, sin embargo, para una conexión asíncrona es posible soportar 432,6 Kbps en ambas direcciones si el enlace es simétrico.

La CPU del dispositivo se encarga de atender las instrucciones relacionadas con Bluetooth del dispositivo anfitrión, para así simplificar su operación. Para ello, sobre la CPU se ejecuta un software denominado *Link manager* que tiene como función comunicarse con otros dispositivos por medio del protocolo LMP.

Entre las tareas realizadas por el control lógico y el manejador de enlace, destacan las siguientes:

- Envío y recepción de datos
- Empaginamiento y peticiones
- Determinación de conexiones
- Autenticación
- Negociación y determinación de tipos de enlace
- Determinación del tipo de cuerpo de cada paquete
- Ubicar del dispositivo

Figura 29. **Comunicación entre elementos vía Bluetooth**



Fuente: *Skills nesp/* http://www.exuberantsolutions.com/lte_core_network_planning_course.htm.
Consulta: 10 de abril de 2015.

2.4.1. Módulos de comunicación Bluetooth

Cuando se piensa en el estándar Bluetooth para comunicar dispositivos; entre estos se hace referencia a objetos como el ordenador, el teléfono móvil o algún dispositivo costoso. Sin embargo, gracias a las facilidades que brindan los módulos de comunicación Bluetooth, se puede, por ejemplo, transformar una simple conexión RS232 en inalámbrica. Con un alcance de hasta 100 metros.

El módulo de desarrollo Bluetooth de la empresa Microingenia S. L, basado en el circuito integrado RN41 de Roving Networks, es un perfecto aliado para eliminar los cables en los proyectos o para conectarse a ordenadores y/o teléfonos móviles. Con un alcance de hasta 100 metros (clase I), posee antena integrada, es compatible con el estándar Bluetooth 2.1, permite velocidades de transferencia de hasta 921Kbps, y lo puede conectar de forma sencilla mediante la USART (RX/TX) de cualquier microcontrolador, desde donde podrá controlarse haciendo uso de sencillos comandos AT. El módulo, además,

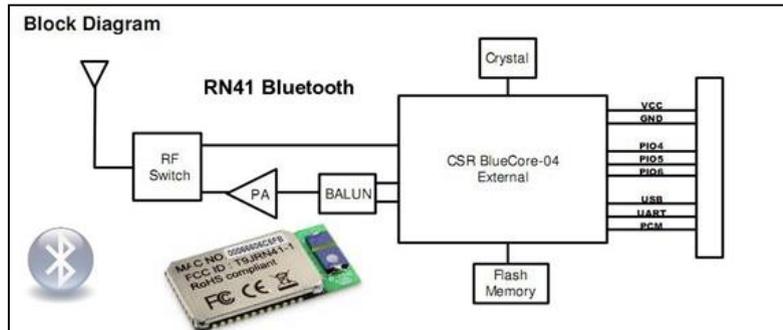
puede ser alimentado tanto a 5 volts como a 3,3 volts, y para ello dispone de un puente de selección de la tensión de alimentación.

El RN41 funciona a 3,3 volts y sus líneas no son 5 volts compatibles, por lo que las líneas de uso normal (Rx/Tx) están provistas de adaptación de niveles para su utilización con las placas de desarrollo que operan a 5Volts. De todos modos, posee entradas auxiliares que te permitirán trabajar en forma directa con el RN41 y un bus Rx/Tx de 3,3 volts.

La posibilidad que brinda utilizar un módulo completo (ya ensamblado) tiene la ventaja de disponer en forma rápida de una conectividad eficiente para proyectos. Es decir, es muy sencillo insertar el módulo en un *protoboard*, alimentarlo con 3,3 volts, conectarle los canales de transmisión y recepción provenientes de la USART de un PIC y agregarle un diodo LED indicador de estado de conexión. El diagrama en bloques del RN41 muestra las posibilidades de trabajo que este módulo ofrece y, como podemos apreciar, posee conexión serie, USB, PCM (audio). Además, dispone de terminales de salida para activar dispositivos diversos como relés, motores, luces y otros circuitos de salida acoplados a este maravilloso enlace inalámbrico que es el módulo RN41.

El circuito utilizado para iniciar la comunicación Bluetooth con un microcontrolador es sencillo, porque solo necesita los elementos básicos. Es el clásico circuito que emplea los componentes más usuales. Un microcontrolador que contenga un módulo USART, con un cristal de 8 Mhz a la velocidad máxima del mismo. Si se utilizan otros microcontroladores con menor velocidad de reloj, se debe observar la conexión de PIO7 en el módulo para forzar la velocidad a 9 600bps.

Figura 30. Diagrama del modulo RN41



Fuente: elaboración propia, empleando Paint.

Es decir, se debe colocar una resistencia *pull-up* (1K) en el mencionado pin respecto de la tensión de alimentación para lograr este objetivo, circuito de alimentación y *reset*.

3. SISTEMA ANDROID

Android es un sistema operativo para dispositivos móviles como teléfonos inteligentes y tabletas basado en el núcleo Linux. Es desarrollado por la Open Handset Alliance, la cual es liderada por Google, usando diversos conjuntos de herramientas de software de código abierto para dispositivos móviles.

Fue construido para permitir a los desarrolladores la creación de aplicaciones móviles que aprovechan al máximo el uso de todas las herramientas que un dispositivo como este puede ofrecer.

Implementa una arquitectura en la que cualquier aplicación puede obtener acceso a las capacidades del teléfono móvil. Por ejemplo, una aplicación puede llamar una o varias de las funcionalidades básicas de los dispositivos móviles, tales como realizar llamadas, enviar mensajes de texto o utilizar la cámara, facilitando a los desarrolladores crear experiencias más ricas y con más coherencia para los usuarios.

Está construido sobre el kernel de Linux. Además, se utiliza una máquina personalizada virtual que fue diseñada para optimizar los recursos de memoria y de hardware en un entorno móvil. Android es de código abierto, y además puede ser libremente ampliado para incorporar nuevas tecnologías de vanguardia que van surgiendo. La plataforma continuará evolucionando a medida que la comunidad de desarrolladores, trabajando juntos, puedan crear aplicaciones móviles innovadoras.

La estructura del sistema operativo Android se compone de aplicaciones que se ejecutan en un *framework* Java de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de Java en una máquina virtual Dalvik con compilación en tiempo de ejecución. Las bibliotecas escritas en lenguaje C incluyen un administrador de interfaz gráfica (*surface manager*), un *frameworkopencore*, una base de datos relacional SQLite, una API gráfica OpenGL ES 2.0 3D, un motor de renderizado *WebKit*, un motor gráfico SGL, SSL y una biblioteca estándar de C Bionic.

El sistema operativo está compuesto por 12 millones de líneas de código, incluyendo 3 millones de líneas de XML, 2,8 millones de líneas de lenguaje C, 2,1 millones de líneas de Java y 1,75 millones de líneas de C++.

3.1. Historia

En el 2003, en Palo Alto, California, fue fundada la compañía Android Inc; la cual centraba sus funciones en el desarrollo de software para teléfonos móviles. En el 2005, la compañía Google al adquirir Android Inc contrató un grupo de los cofundadores de la misma, entre los que se incluían Andy Rubin, Rich Miner, Nick Sears y Chris White. El equipo que encabezó Rubin desarrolló una plataforma para dispositivos móviles que fue anunciada a fabricantes de dispositivos y operadores bajo la promesa de dotar de un sistema actualizable y flexible. Este sistema estaba basado en el *kernel* de Linux.

En el 2006 ya existían rumores acerca de que Google tenía intención de entrar en el mercado de la telefonía móvil, ya que la empresa se empeñaba en que sus servicios de búsqueda y aplicaciones estuvieran disponibles para estos medios.

Para el 2007, se difundió que la empresa estaba solicitando patentes en telefonía móvil. El 5 de noviembre de 2007 se creó con el fin de desarrollar estándares abiertos para dispositivos móviles la Open Handset Alliance (OHA), un consorcio de varias compañías dedicadas a la telefonía, software, comercialización, semiconductores y fabricantes de electrónicos. Junto con la formación de la OHA, el 12 de noviembre se estrenó su primer producto, el Android Software Development Kit (SDK), una plataforma para dispositivos móviles construida sobre la versión 2.6 del *kernel* de Linux.

Es hasta el 22 de octubre de 2008 cuando sale al mercado el primer teléfono que ejecuta Android.

3.2. Arquitectura

La arquitectura del sistema operativo está compuesta por cuatro capas; la primera de ellas es un kernel basado por Linux, le siguen las bibliotecas entre las que se encuentran las básicas correspondientes a la máquina virtual, a continuación está el marco de aplicaciones o *framework* y finalmente las aplicaciones.

3.2.1. Núcleo Linux

Android se basa en Linux para los servicios base del sistema como gestión de memoria y de procesos, pila de red, modelo de controladores y seguridad. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software. Esto permite que se pueda acceder a los componentes sin necesidad de conocer el modelo o características precisas que están instaladas en cada dispositivo.

3.2.2. Bibliotecas

Esta capa incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas bibliotecas ofrecen a los desarrolladores a través del marco de trabajo de aplicaciones Android; algunas son: System C library (implementación de biblioteca C estándar), bibliotecas de medios, bibliotecas de gráficos, 3D y SQLite, entre otras.

Tabla I. **Bibliotecas del sistema Android**

| Biblioteca | Función |
|---|--|
| Gestor de superficies(<i>Surface manager</i>) | Se encarga de componer las imágenes que se muestran en la pantalla a partir de capas gráficas 2D y 3D. Cada vez que la aplicación pretende dibujar algo en la pantalla, la biblioteca no lo hace directamente sobre ella, en vez de eso realiza los cambios en imágenes (mapas de Bits) que almacenan en memoria y que después combinan para formar la imagen final que se envía a pantalla. Esto permite realizar con facilidad diversos efectos: superposición de elementos, transparencias, transiciones, animaciones, entre otros. |
| SGL (<i>Scalable graphics library</i>) | Desarrollada por Skia (empresa adquirida por Google en 2005) y utilizada tanto en Android como en Chrome; se encarga de representar elementos en dos dimensiones. Es el motor gráfico 2D de Android. |
| OpenGL ES (<i>Open GL for embedded systems</i>) | Motor gráfico 3D basado en las APIS (<i>Application program interface</i>) de OpenGL ES 1.0, 1.1 (desde la versión 1.6 de Android) y 2.0 (desde la versión 2.2 de Android). Utiliza aceleración hardware (si el teléfono la proporciona) o un motor software altamente optimizado (según Google) cuando no la hay. |

Continuación de tabla I.

| | |
|-------------------------------------|--|
| Bibliotecas multimedia | Basadas en Open-CORE, permiten visualizar, reproducir e incluso grabar numerosos formatos de imagen, video y audio como JPG, GIF, PNG, MPEG4, AVC (H.264), MP3, AAC o AMR. |
| WebKit | Motor web utilizado por el navegador (tanto como aplicación independiente como embebido en otras aplicaciones). Es el mismo motor que utilizan Google Chrome y Safari (el navegador de Apple, tanto en Mac como en el iPhone). |
| SSL (<i>Secure Sockets Layer</i>) | Proporciona seguridad al acceder a Internet por medio de criptografía. <i>Free type</i> : permite mostrar fuentes tipográficas, tanto basadas en mapas de bits como vectoriales. |
| SQLite | Motor de bases de datos relacionales, disponible para todas las aplicaciones. |
| Biblioteca C de sistema (libc) | Basada en la implementación de Berkeley Software Distribution (BSD), pero optimizada para sistemas Linux embebidos. Proporciona funcionalidad básica para la ejecución de las aplicaciones. |
| <i>Runtime</i> de Android | Bibliotecas de entorno de ejecución (no se considera una capa en sí mismo, dado que también está formado por bibliotecas), Android incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas habituales del lenguaje Java. Cada aplicación del sistema corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Esta ha sido escrita de forma que un dispositivo puede correr múltiples máquinas virtuales eficientemente. La máquina virtual ejecuta archivos en el formato Dalvik Executable (.dex), el cual está optimizado para un uso de memoria mínimo. |

Fuente: elaboración propia.

3.2.3. Marco de trabajo de aplicaciones

La arquitectura está diseñada para simplificar la reutilización de componentes; una aplicación puede publicar sus capacidades y después otra aplicación puede hacer uso de las mismas, siempre que atienda a las reglas de seguridad del *framework* usados por las aplicaciones base.

La mayoría de los componentes de esta capa son bibliotecas Java que acceden a los recursos a través de la máquina virtual Dalvik. Entre las más importantes se encuentran las siguientes: el administrador de actividades, administrador de ventanas, proveedor de contenidos, vistas, administrador de notificaciones, de paquetes, de telefonía, de recursos, de ubicaciones, de sensores, cámara y multimedia.

Tabla II. **Bibliotecas Java**

| Biblioteca | Función |
|---|---|
| Administrador de actividades (<i>activity manager</i>) | Se encarga de controlar el ciclo de vida de las actividades y la propia pila de las mismas. |
| Administrador de ventanas (<i>Windows manager</i>) | Se encarga de organizar lo que se muestra en pantalla, creando superficies que pueden ser “rellenadas” por las actividades. |
| Proveedor de contenidos (<i>Content provider</i>) | Permite encapsular un conjunto de datos que se compartirá entre aplicaciones creando una capa de abstracción que hacen accesibles dichos datos, sin perder el control sobre cómo se accede a la información. Por ejemplo, uno de los proveedores de contenido existentes permite a las aplicaciones acceder a los contactos almacenados en el teléfono. |

Continuación de la tabla II.

| | |
|---|---|
| Vistas (<i>Views</i>) | Si antes se equiparaban las actividades con las ventanas de un sistema operativo de PC, las vistas se pueden comparar con los controles que se suelen incluir dentro de esas ventanas. Android proporciona numerosas vistas con las que construir las interfaces de usuario: botones, cuadros de texto, listas, entre otros. También proporciona otras más sofisticadas, como un navegador web o un visor de Google Maps. |
| Administrador de notificaciones (<i>Notification manager</i>) | Proporciona servicios para notificar al usuario cuando algo requiera su atención. Normalmente las notificaciones se realizan mostrando alerta en la barra de estado, pero esta biblioteca también permite emitir sonidos, activar el vibrador o hacer parpadear los LEDs del teléfono (si los tiene). |
| Administrador de paquetes (<i>Package manager</i>) | Las aplicaciones Android se distribuyen en paquetes (archivos .apk) que contienen tanto los archivos .dex como todos los recursos y archivos adicionales que necesite la aplicación, para facilitar su descarga e instalación. Esta biblioteca permite obtener información sobre los paquetes actualmente instalados en el dispositivo Android, además de gestionar la instalación de nuevos paquetes. |
| Administrador de sensores (<i>sensor manager</i>) | Permite gestionar todos los sensores, hardware disponibles en el dispositivo Android: acelerómetro, giroscopio, sensor de luminosidad, sensor de campo magnético, brújula, sensor de presión, sensor de proximidad, sensor de temperatura, entre otros. |
| Administrador de telefonía (<i>Telephony manager</i>) | Proporciona acceso a la pila hardware de telefonía del dispositivo Android, si la tiene. Permite realizar llamadas o enviar y recibir SMS/MMS, aunque no permite reemplazar o eliminar la actividad que se muestra cuando una llamada está en curso (por motivos de seguridad). |

Continuación de la tabla II.

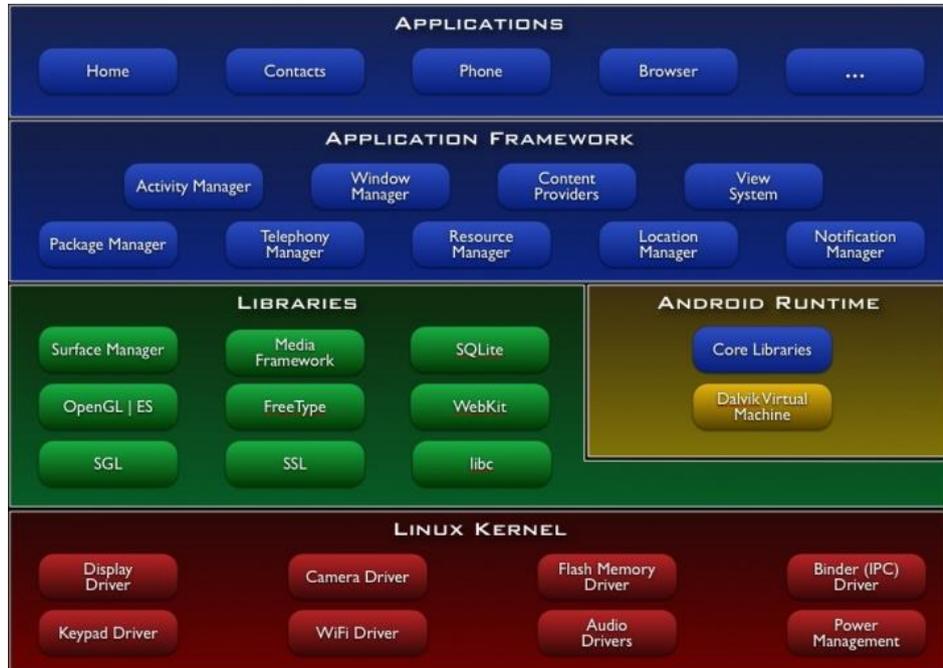
| | |
|--|--|
| Administrador de recursos (<i>Resource manager</i>) | Proporciona acceso a todos los elementos propios de una aplicación que se incluyen directamente en el código: cadenas de texto traducidas a diferentes idiomas, imágenes, sonidos e incluso disposiciones de las vistas dentro de una actividad (<i>layouts</i>). Permite gestionar esos elementos fuera del código de la aplicación y proporcionar diferentes versiones, por ejemplo en función del idioma del dispositivo o la resolución de pantalla que tenga. |
| Administrador de ubicaciones (<i>Location manager</i>) | Permite determinar la posición geográfica del dispositivo Android (usando el GPS o las redes disponibles) y trabajar con mapas. |
| Cámara | Proporciona acceso a las cámaras del dispositivo Android, tanto para tomar fotografías como para grabar video. |
| Multimedia | Conjunto de bibliotecas que permiten reproducir y visualizar audio, video e imágenes en el dispositivo. |

Fuente: elaboración propia.

3.2.4. Aplicaciones

Las aplicaciones base incluyen un navegador, cliente de correo electrónico, programa de mensajería, calendario, mapas, contactos, juegos y el inicio (*launcher*) que es donde se muestran las aplicaciones instaladas, permitiendo lanzarlas así como *widgets* (pequeñas aplicaciones). Las aplicaciones están escritas en Java o C/C++.

Figura 31. **Arquitectura del sistema Android**



Fuente: Laboratorio de software libre. http://1.bp.blogspot.com/-Hd1FDyPFLqw/UP8oIOctTOI/AAAAAAAAABY/2AOyHQJ_Sq8/s1600/img1.png. Consulta: 30 de abril de 2015.

3.3. **Versiones del sistema Android**

El sistema Android ha visto numerosas actualizaciones desde su liberación inicial. Estas actualizaciones al sistema operativo base típicamente arreglan *bugs* y agregan nuevas funciones. A cada versión de Android se le asocia en inglés un nombre de postre, esto a partir de la versión 1.5. El postre elegido empieza por una letra distinta siguiendo un orden alfabético.

En la tabla III se enlista la fecha de salida y características más importantes.

Tabla III. **Versiones del sistema operativo Android**

| Versiones | Fecha de liberación y características |
|--|---|
| 1.0 | <ul style="list-style-type: none"> • Liberado el 23 de septiembre de 2008. |
| 1.1 | <ul style="list-style-type: none"> • Liberado el 9 de febrero de 2009. |
| <p>1.5 (<i>Cupcake</i>) basado en el <i>kernel</i> de Linux 2.6.27</p> | <p>El 30 de abril de 2009, la actualización 1.5 (<i>Cupcake</i>) para Android fue liberada. Hubo varias características nuevas y actualizaciones en la interfaz de usuario en la</p> <p>la actualización 1.5:</p> <ul style="list-style-type: none"> • Posibilidad de grabar y reproducir videos a través del modo <i>camcorder</i>. • Capacidad de subir videos a YouTube e imágenes a Picasa directamente desde el teléfono. • Un nuevo teclado con predicción de texto. • Soporte para Bluetooth A2DP y AVRCP. • Capacidad de conexión automática para conectar a auricular Bluetooth a cierta distancia. • Nuevos <i>widgets</i> y carpetas que se pueden colocar en las pantallas de inicio. • Transiciones de pantalla animadas. |

Continuación de tabla III.

| | |
|---|---|
| <p>1.6 (<i>Donut</i>) basado en el kernel de Linux 2.6.29</p> | <p>El 15 de septiembre de 2009, el SDK 1.6 (<i>Donut</i>) fue liberado. Se incluyó en esta actualización:</p> <ul style="list-style-type: none">• Una experiencia mejorada en el AndroidMarket.• Una interfaz integrada de cámara, filmadora y galería.• La galería ahora permite a los usuarios seleccionar varias fotos para eliminarlas.• Búsqueda por voz actualizada, con respuesta más rápida y mayor integración con aplicaciones nativas, incluyendo la posibilidad de marcar a contactos.• Experiencia de búsqueda mejorada que permite buscar marcadores, historiales, contactos y páginas web desde la pantalla de inicio.• Actualización de soporte para CDMA/EVDO, 802.1x, VPN y <i>text-to-speech</i>.• Soporte para resoluciones de pantalla WVGA.• Mejoras de velocidad en las aplicaciones de búsqueda y cámara.• <i>Framework</i> de gestos y herramienta de desarrollo <i>GestureBuilder</i>.• Navegación gratuita <i>turn-by-turn</i> de Google. |
|---|---|

Continuación de tabla III.

| | |
|---|---|
| <p>2.0 / 2.1 (Eclair) basado en el kernel de Linux 2.6.29</p> | <p>El 26 de octubre de 2009, el SDK 2.0 (Eclair) fue liberado. Los cambios incluyeron:</p> <ul style="list-style-type: none">• Velocidad de hardware optimizada.• Soporte para más tamaños de pantalla y resoluciones.• Interfaz de usuario renovada.• Nuevo interfaz de usuario en el navegador y soporte para HTML5.• Nuevas listas de contactos.• Una mejor relación de contraste para los fondos.• Mejoras en Google Maps 3.1.2.• Soporte para Microsoft Exchange.• Soporte integrado de <i>flash</i> para la cámara.• Zoom digital.• <i>MotionEvent</i> mejorado para captura de eventos <i>multitouch</i>.• Teclado virtual mejorado.• Bluetooth 2.1.• Fondos de pantalla animados.• El SDK 2.0.1 fue liberado el 3 de diciembre de 2009.• El SDK 2.1 fue liberado el 12 de enero de 2010. |
|---|---|

Continuación de tabla III.

| | |
|---|---|
| <p>2.2 (<i>Froyo</i>) basado en el kernel de Linux 2.6.32</p> | <p>El 20 de mayo de 2010, el SDK 2.2 (<i>Froyo</i>) fue liberado. Los cambios incluyeron:</p> <ul style="list-style-type: none">• Optimización general del sistema Android, la memoria y el rendimiento.• Mejoras en la velocidad de las aplicaciones, gracias a la implementación de JIT.• Integración del motor JavaScript V8 del Google Chrome en la aplicación Browser.• Soporte mejorado de Microsoft Exchange (reglas de seguridad, reconocimiento automático, <i>GAL look-up</i>, sincronización de calendario, limpieza remota).• Lanzador de aplicaciones, mejorado, con accesos directos a las aplicaciones de teléfono.• Funcionalidad de <i>Wi-Fi hotspot</i> y <i>tethering</i> por USB.• Permite desactivar el tráfico de datos a través de la red del operador.• Actualización del <i>market</i> con actualizaciones automáticas.• Cambio rápido entre múltiples idiomas de teclado y sus diccionarios.• Marcación por voz y compartir contactos por Bluetooth. |
|---|---|

Continuación de tabla III.

| | |
|--|---|
| <p>2.3 (<i>Gingerbread</i>) basado en el kernel de Linux 2.6.35.7 Actual en smat</p> | <p>El 6 de diciembre de 2010, el SDK 2.3 (<i>Gingerbread</i>) fue liberado. Los cambios incluyeron:</p> <ul style="list-style-type: none">• Soporte para dispositivos móviles.• Actualización del diseño de la interfaz de usuario.• Soporte para pantallas extra grandes y resoluciones WXGA y mayores.• Soporte nativo para telefonía <i>VoIP SIP</i>.• Soporte para reproducción de videos WebM/VP8 y decodificación de audio AAC.• Nuevos efectos de audio como reverberación, ecualización, virtualización de los auriculares y refuerzo de graves.• Soporte para <i>Near Field Communication</i>.• Funcionalidades de cortar, copiar y pegar disponibles a lo largo del sistema.• Teclado multi-táctil rediseñado.• Soporte mejorado para desarrollo de código nativo.• Mejoras en la entrada de datos, audio y gráficos para desarrolladores de juegos.• Recolección de elementos concurrentes para un mayor rendimiento. |
|--|---|

Continuación de tabla III.

| | |
|--|--|
| <p>3.0 / 3.1 / 3.2 (Honeycomb)</p> | <ul style="list-style-type: none">• Mejor soporte para <i>tablets</i>.• Escritorio 3D con <i>widgets</i> rediseñados.• Sistema multitarea mejorado.• Mejoras en el navegador <i>web</i> predeterminado, entre lo que destaca la navegación por pestañas, autorelleno de formularios, sincronización de favoritos con Google Chrome y navegación privada.• Soporte para video chat mediante Google Talk.• Mejor soporte para redes <i>Wi-Fi</i>.• Añade soporte para una gran variedad de periféricos y accesorios con conexión USB: teclados, ratones, <i>hubs</i>, dispositivos de juego y cámaras digitales. Cuando un accesorio está conectado, el sistema busca la aplicación necesaria y ofrece su ejecución.• Los <i>widgets</i> pueden redimensionarse de forma manual sin la limitación del número de cuadros que tenga cada escritorio.• Se añade soporte opcional para redimensionar correctamente las aplicaciones inicialmente creadas para móvil para que se vean bien en <i>tablets</i>. |
|--|--|

Continuación de tabla III.

| | |
|--|---|
| <p>4.0 (<i>Ice Cream Sandwich</i>)</p> | <ul style="list-style-type: none">• Versión que unifica el uso en cualquier dispositivo, tanto en teléfonos, <i>tablets</i>, televisores, <i>netbooks</i>, entre otros.• Interfaz limpia y moderna con una nueva fuente llamada "<i>Roboto</i>", muy al estilo de <i>Honeycomb</i>.• Opción de utilizar los botones virtuales en la interfaz de usuario, en lugar de los botones táctiles capacitivos.• Llega la aceleración por hardware, lo que significa que la interfaz podrá ser manejada y dibujada por la GPU y aumentando notablemente su rapidez, su respuesta y evidentemente, la experiencia de usuario.• Multitarea mejorada, estilo <i>Honeycomb</i>, añadiendo la posibilidad de finalizar una tarea simplemente desplazándola fuera de la lista.• Ha añadido un gestor del tráfico de datos de internet. El entorno le permite establecer alertas cuando llegue a una cierta cantidad de uso y desactivación de los datos cuando se pasa de su límite.• Los <i>widgets</i> están en una nueva pestaña, que figuran en una lista similar a las aplicaciones en el menú principal. |
|--|---|

Continuación de tabla III.

| | |
|--------------------------|--|
| 4.0 (Ice Cream Sandwich) | <ul style="list-style-type: none">• El corrector de texto ha sido rediseñado y mejorado, ofreciendo la opción de tocar en una palabra para que aparezca una lista con las diferentes opciones de edición y sugerencias de palabras similares.• Las notificaciones tiene la posibilidad de descartar las que no son importantes y también desplegar la barra de notificaciones con el dispositivo bloqueado.• La captura de pantalla, con solo pulsar el botón de bajar volumen y el botón de encendido.• La aplicación de la cámara se ha llevado un buen lavado de cara, con nuevas utilidades, como la posibilidad de hacer fotografías panorámicas de forma automática.• AndroidBeam es la nueva característica que permitirá compartir contenido entre teléfonos. Vía NFC (<i>Near Field Communication</i>).• Reconocimiento de voz del usuario.• Aplicación de teléfono nuevo con la funcionalidad de buzón de voz visual, que le permite adelantar o retroceder los mensajes de voz. |
|--------------------------|--|

Continuación de tabla III.

| | |
|--------------------------|--|
| 4.0 (Ice Cream Sandwich) | <ul style="list-style-type: none">• Reconocimiento facial, lo que haría que se pueda cambiar la vista.• Las carpetas son mucho más fáciles de crear, con un estilo de arrastrar y soltar.• Un único y nuevo <i>framework</i> para las aplicaciones.• El usuario tendrá herramientas para ocultar y controlar las aplicaciones que “cuelgue” la operadora de turno o el fabricante, liberando recursos de segundo plano (ciclos de ejecución y memoria Ram). No obstante, no se podrán desinstalar.• Soporte nativo del contenedor MKV.• Soporte nativo para el uso de <i>Stylus</i> (lápiz táctil). |
|--------------------------|--|

Fuente: elaboración propia.

3.4. Equipos que utilizan sistema Android

El sistema operativo Android inicialmente se usó en teléfonos; sin embargo a partir de la versión 2.0 se extendió a ordenadores portátiles, *netbooks*, *tablets*, Google TV, relojes de pulsera, auriculares y otros dispositivos (electrodomésticos inclusive). La plataforma de hardware principal de Android es la arquitectura ARM, aunque como ya se mencionó, hay soporte para x86 en el proyecto Android-x86; y Google TV utiliza una versión especial de Android x86, también se han desarrollado dispositivos externos para ejecutar Android

4.0 en un televisor que tenga una conexión HDMI y un puerto USB para su alimentación.

3.4.1. Smartphone

Se denomina *smartphone* a la familia de teléfonos móviles que disponen de un hardware y un sistema operativo propio capaz de realizar tareas y funciones similares a las realizadas por los ordenadores fijos o portátiles, añadiéndole al teléfono funcionalidades extras a la realización y recepción de llamadas y mensajes telefónicos. Conocidos también como teléfonos inteligentes (*smart* es inteligente y *phone* es teléfono en inglés) son considerados como la evolución tecnológica a los clásicos teléfonos móviles.

Los primeros smartphones se diferenciaron de los móviles de la época añadiendo funciones extras como organizadores personales incorporados en el teléfono; estos organizadores incluían aplicaciones como un bloc de notas, un calendario dónde anotar citas, reuniones y alarmas, un gestor para la recepción y envío de correos electrónicos (email), un teclado Qwerty que facilitaba la escritura en el teléfono, entre otros. Estos móviles dieron un paso tecnológico con el objetivo de asemejarse a ciertas funcionalidades que solo los ordenadores fijos y portátiles de aquella época podían ejecutar, pero con la ventaja de tenerlo en un pequeño dispositivo fácilmente transportable.

Con el tiempo y el desarrollo tecnológico de los últimos años, los Smartphone disponibles hoy en día poseen una serie de características y funcionalidades extras que les diferencian claramente de los móviles convencionales, funcionalidades como:

- Disponen de una aplicación para el envío y recepción de correo electrónico, así como la gestión de varias cuentas de correo.
- Disponen de una suite de aplicaciones focalizadas a realizar funciones de organizador personal como calendarios, recordatorios y alertas y block de notas. los cuales pueden comunicarse y sincronizarse con otros ordenadores, *tablets* o móviles.
- Disponen de una conexión a internet, gracias a la red 3G y 4G, que permite navegar por la red al igual que si se accediese desde un ordenador fijo.
- Pueden leer, editar y reproducir una amplia familia de archivos como hojas de cálculo, editores de textos, archivos multimedia de video y música, entre otros.
- Permiten la descarga y la ejecución de aplicaciones (APP) desarrollados por terceros los cuales amplían nuevas funcionalidades, por ejemplo juegos, retoques fotográficos, lectores de libros electrónicos y navegadores GPS.
- Disponen de un teclado Qwerty físico o táctil, el cual permite y facilita la escritura de datos en el teléfono.

Por último y más importante, disponen de un sistema operativo capaz de desarrollar todas las funcionalidades descritas anteriormente. iOS de la empresa Apple, Android de la empresa Google o Windows phone por parte de la empresa Microsoft, son ejemplos de sistemas operativos diseñados y

programados para hacer trabajar a los *smartphone*, como un ordenador de bolsillo.

Además de las funcionalidades descritas anteriormente, hoy en día los fabricantes de *smartphone* incluyen en sus modelos cámaras de alta resolución que permiten realizar fotografías, grabar vídeos en HD y videoconferencias, receptores GPS que permiten conocer con total exactitud la localización y trayecto de cualquier punto del mundo, giroscopios, acelerómetros, sensores de luminosidad y proximidad, así como sensores para la identificación de huellas dactilares.

Los *smartphone* parten del principio por el cual el teléfono no sirve solo para llamar y escribir mensajes; ahora en el teléfono se puede consultar cualquier página de internet, leer y editar cualquier hoja de cálculo o editor de texto, descargar y reproducir una película o un álbum de música, utilizarlo como navegador GPS mientras se conduce, fotografiar o realizar un video en alta definición en cualquier momento, controlar la televisión del salón como si fuera un mando a distancia. Ahora en el móvil se tiene un ordenador con una gran potencia de cálculo en un espacio reducido y fácilmente transportable.

3.4.2. Tablet

Una tablet es un dispositivo electrónico que no necesita estar conectado continuamente a la red eléctrica, para funcionar, es decir es totalmente móvil. Posee además una pantalla táctil con la cual se pueden manejar casi todos los aspectos del sistema y sujetarse con una sola mano.

Entre sus funciones está la capacidad para navegar por internet, dando en este caso una experiencia de usuario muy parecida a la de cualquier ordenador

moderno; se puede consultar correo electrónico, reproducir películas, oír música, ver fotos y jugar; también funciona como GPS, o para leer libros. En esencia, es capaz de hacer el 80 % de las tareas que realiza una persona normal con un ordenador personal.

A sus características se añade un arranque casi instantáneo, un menor precio, y que su uso y configuración es mucho más sencillo que el de un PC; allí están todas las razones de su éxito.

Lo que se entiende como tablet en la actualidad surge de la llegada del primer iPad de Apple en 2010, el cual supone un salto cualitativo en el uso de internet por parte de estos dispositivos. Por primera vez se tenía una experiencia parecida a la navegación en un ordenador en la palma de la mano.

Pero es imposible entender la revolución posterior y en la que todavía se está inmerso sin hablar del sistema operativo Android desarrollado por Google. Este ha permitido poner a los montadores de estos dispositivos en el mercado toda clase de tablets, pensadas desde aquellos que no necesitan equipos muy potentes y tienen presupuestos bajos, hasta los que prefieren comprar lo último de lo último y no están tan preocupados por el precio.

Las tablets cuentan con utilidades de varios tipos entre las cuales están:

- Lectura de libros electrónicos
- Consulta y edición de documentos ofimáticos
- Navegación web (mediante wifi o USB)
- Llamadas telefónicas, si son 3G, sustituyendo así al teléfono móvil
- GPS
- Reproducción de música

- Visualización de vídeos y películas
- Cámara fotográfica
- Videoconferencia
- Juegos

3.4.3. Netbook

El concepto de *netbook*, que fue incluido para una gama de ordenadores que llevaba una configuración de hardware bastante similar al de una *handheld*, siendo utilizado en 1999 en primer lugar por la firma Psion, siendo reintroducido en el mercado en el 2008 por la compañía Intel, con una aplicación completamente diferente.

Tal como se conoce ahora, los *netbooks* son equipos bastante similares a las UMPC (es decir, PC ultra portátil) pero con un diseño más grande, generalmente oscilando entre las 7 y las 14 pulgadas de diagonal en su pantalla, y con un peso que a duras penas alcanza los dos kilogramos, lo que facilita muchísimo su portabilidad, llevándola en un pequeño bolso de mano o simplemente en una mochila, lo que hace que muchas escuelas las incluyan como parte de su material didáctico.

Este ahorro de dimensiones y peso se logra utilizando una configuración de hardware inferior a la de otros equipos, contando con una menor potencia y un rendimiento no tan elevado, además de carecer de puertos no esenciales, y la que es considerada como una característica determinante: no poseen unidades ópticas (es decir, unidad de CD, DVD o Blu-Ray).

Pueden utilizar tanto los sistemas operativos basados en GNU o Linux, tanto como aquellos que son distribuidos por la compañía Microsoft, aunque a

veces por cuestiones de costos se incluyen aquellos con licencias gratuitas, como por ejemplo el Microsoft Windows XP, aunque también se tiene aquellos que han sido preparados exclusivamente para sacar el máximo provecho de estos equipos, como el Ubuntu Netbook Remix, aunque hoy en día hay muchos equipos que incluyen una versión del Microsoft Windows 7 Starter Edition. Un nuevo dispositivo que ha lanzado la empresa Go Electronics, es una Netbook de 10 pulgadas llamado GoNote que presenta la novedad de incorporar Android 4.0 (*Ice Cream Sandwich*).

3.4.4. Smart TV

Smart TV es una denominación que las compañías de televisores han dado a sus modelos conectados y más avanzados. En ellos, además de posibilidad de reproducir contenido desde diferentes fuentes, coincide una conexión a Internet que puede ser vía wifi y la posibilidad de instalar o usar aplicaciones diseñadas específicamente para un televisor.

Esos televisores Smart TV también facilitan la comunicación, el acceso a las redes sociales y suelen integrar un buscador para localizar contenido entre las múltiples fuentes de las que disponen. En los modelos de 2012 se está introduciendo también el control por voz y gestos.

Todas las marcas actuales que poseen televisores en el mercado ofrecen modelos que se pueden considerar como Smart TV, aunque la interoperabilidad de sus aplicaciones y servicios no está para nada conseguida.

Si no se quiere adquirir un nuevo televisor, casi todas las marcas que tienen plataformas Smart TV propias integran esa funcionalidad de forma completa en otros equipos que se coloca al lado del televisor, como centros

multimedia o reproductores de salón tal como los Blu-Ray. Bastará pues con hacerse con uno y vía HDMI, convertir un televisor no Smart TV en uno conectado.

3.4.4.1. Android TV

TV Android es una nueva variante del sistema operativo móvil de Google dirigida a televisores, consolas de videojuegos y decodificadores. El sucesor del malogrado Google TV, Android TV, trae una nueva interfaz de usuario optimizada para televisores y decodificadores, diseñados para poner delante contenido y centro. Se basa en Android 5.0 Lollipop, que por supuesto también funciona en teléfonos, tabletas y coches a través de Android Auto.

Con Android TV se podrá ver el contenido de una variedad de proveedores como Google Play, Netflix y YouTube, así como jugar una serie de juegos para Android. La búsqueda por voz de Google ayudará a conseguir lo que se quiere con facilidad y con Google moldeada integrada en todas las aplicaciones que soportan Chromecast será capaz de interactuar con Android TV. La interfaz se controla principalmente a través de controles remotos de estilo televisivo incluidos, pero también se puede utilizar con un dispositivo de juego de Bluetooth o una aplicación de control remoto en su teléfono Android.

Los desarrolladores de aplicaciones pueden dirigirse a Android TV específicamente en el desarrollo, y las aplicaciones y juegos de diseño con una interfaz que está diseñada específicamente para ser visto desde una distancia y controlado con un mando a distancia o del juego.

Sony se ha comprometido a poner Android TV en su completa línea de TV del 2015, con Sharp y TPVision, se comprometieron a utilizar el sistema Android

TV con su línea de televisores 2015. TV Android, y el primer hardware para ir con este sistema; fueron anunciados en Google I / O 2014. Desde entonces Google también lanzó el Nexus Player, su primer equipo de cara al cliente para ejecutar Android TV; y otros fabricantes han anunciado un pequeño decodificador con el software a bordo.

3.4.5. Smartwatch

Es un reloj inteligente. La palabra, como tal, es un vocablo inglés compuesto por las palabras *smart*, que significa ‘inteligente’, y *watch*, que traduce ‘reloj’.

Smartwatch, en este sentido, es un nombre comercial que se ha venido empleando para designar productos de alta tecnología, como teléfonos celulares (*smartphone*) o televisores (*smart TV*), y que se distinguen, entre otras cosas, por tener la capacidad para funcionar en red y proporcionar acceso a internet.

El *smartwatch* evoluciona del reloj de pulsera digital para incorporar una serie de funciones y opciones diseñadas para satisfacer las necesidades y exigencias de los usuarios contemporáneos.

De allí que *smartwatch* sea comparable a una agenda personal digital o una computadora portátil, con el agregado de que puede también funcionar como un teléfono, pues uno de sus objetivos es facilitar la interacción con el *smartphone*.

En este sentido, los *smartwatches* permiten al usuario operar con tecnologías inalámbricas, como wifi, Bluetooth y GPS, ejecutar aplicaciones, así

como guardar, recuperar y sincronizar datos. Además, posee funciones que van desde lo más básico, como calculadora, acelerómetro, altímetro, barómetro, giroscopio, brújula, cronógrafo o desde luego, reloj; hasta lo más complejo, como traductor, juegos, GPS, agenda personal, teléfono (incluye micrófono, altavoz, o auriculares inalámbricos o manos libres), así como una interfaz gráfica que facilita la interacción con el aparato.

No obstante, el *smartwatch* ha sido concebido para servir de interfaz de un sistema remoto, principalmente de teléfonos inteligentes. En este sentido, pueden ser considerados una extensión del *smartphone*, aunque se pronostica que en un futuro próximo, el *smartwatch* pueda contener independientemente todas las funciones de un teléfono inteligente.

3.5. Aplicaciones Android

Las aplicaciones incluidas por defecto de Android serán aquellas que el usuario vaya añadiendo posteriormente, ya sea de terceras empresas o de su propio desarrollo. Todas estas aplicaciones utilizan los servicios, las API y librerías de los niveles anteriores.

En la última capa se incluyen todas las aplicaciones del dispositivo, tanto las que tienen interfaz de usuario como las que no, las nativas (programadas en C o C++) y las administradas (programadas en Java), las que vienen preinstaladas en el dispositivo y aquellas que el usuario ha instalado.

En esta capa se encuentra también la aplicación principal del sistema: Inicio (*Home*) o lanzador (*Launcher*), porque es la que permite ejecutar otras aplicaciones mediante una lista y mostrando diferentes escritorios donde se

pueden colocar accesos directos a aplicaciones o incluso *widgets*, que son también aplicaciones de esta capa.

Una aplicación Android corre dentro de su propio proceso Linux, por tanto, una característica fundamental es que el tiempo y ciclo de vida de una aplicación no está controlada por la misma aplicación, sino que lo determina el sistema a partir de una combinación de estados, lo cuales pueden ser las aplicaciones que están funcionando, la prioridad que tiene el usuario y la memoria que queda disponible en el sistema.

Una aplicación en Android debe declarar todas sus actividades, los puntos de entrada, la comunicación, las capas, los permisos, y las intenciones a través de `AndroidManifest.xml`. Es muy importante tener en consideración cómo estos componentes impactan en el tiempo de vida del proceso asociado con una aplicación, porque si no son empleados de manera apropiada, el sistema detendrá el proceso de la aplicación aun cuando se esté haciendo algo importante.

3.5.1. *Framework* de aplicaciones

Representa fundamentalmente el conjunto de herramientas de desarrollo de cualquier aplicación. Toda aplicación que se desarrolle para Android, ya sean las propias del dispositivo, las desarrolladas por Google o terceras compañías, o incluso las que el propio usuario cree, utilizan el mismo conjunto de API y el mismo "*framework*", representado por este nivel.

3.5.2. Java

Java es un lenguaje orientado a objetos que alcanzó su madurez con la popularización de internet y que es, en cierta manera, el heredero legítimo de C++. La expansión de este lenguaje entre la comunidad de programadores ha sido vertiginosa y se ha impuesto como el paradigma de los lenguajes de programación orientados a objetos. En el entorno académico y de investigación, la enseñanza de Java ha remplazado (y está remplazando) a la enseñanza de lenguajes de programación estructurada como Pascal e incluso C, que siempre se han considerado lenguajes de elección para la introducción a la programación.

De forma resumida, Java es un lenguaje neutral, portable, robusto, estable, independiente de la plataforma, sencillo de aprender para programadores que hayan trabajado previamente con lenguajes orientados a objetos. Java puede utilizarse para realizar aplicaciones en múltiples plataformas hardware y sistemas operativos (Unix, Linux, OS/390, Windows, o HP-UX entre otros sistemas operativos para ordenadores personales o estaciones de trabajo, Android, Palm OS o EPOC, entre otros sistemas operativos para dispositivos de telefonía móvil).

Una de las novedades revolucionarias que introdujo Java es la portabilidad. Sun abordó el problema introduciendo el modelo de *bytecode*, cuando un programa Java se compila no se transforma en un conjunto de instrucciones en código-máquina nativas de la plataforma utilizada, lo cual impediría su completa portabilidad, sino que se transforma en un conjunto de *bytecodes* independientes de la plataforma utilizada, que son leídos e interpretados por la máquina virtual Java, JVM, para ejecutar el programa. Por ejemplo, cuando se compila un programa Java en una plataforma

Windows/Intel, se obtiene la misma salida compilada, el mismo bytecode, que en un sistema Macintosh o Unix.

Los requisitos de desarrollo para Android exigen el uso del JDK en su versión 6. El desarrollo para aplicaciones Android se realiza de forma común en Java, aunque existe la posibilidad de realizar parte de una aplicación mediante otros lenguajes utilizando el set de herramientas NDK (*Native Development Kit*), aunque no es recomendable debido a que aumenta la complejidad del desarrollo de las aplicaciones.

3.5.3. Software para realizar aplicaciones Android

El desarrollo de aplicaciones Android se hace habitualmente con lenguaje de programación Java y el conjunto de herramientas de desarrollo SDK, pero hay otras opciones disponibles que pueden ayudar para el diseño de una aplicación Android.

3.5.3.1. SDK

SDK son las siglas *de Software Development Kit*, lo que significa un kit de desarrollo de software. Mediante este kit se pueden desarrollar aplicaciones y ejecutar un emulador de la versión de Android. En Android todas las aplicaciones se ejecutan en Java.

Si se quisiera desarrollar alguna aplicación para Android, para eso se necesita instalar el Android SDK. Para Android se puede conseguir el kit de desarrollo desde su web oficial. Una vez se haya descargado, simplemente debe ejecutarse el programa.

Al haberse descargado e instalado, podrán instalarse en él distintas versiones de Android; para el emulador se realizarán todos los pasos necesarios para configurarlo. Cada vez que aparece una nueva versión de Android, Google libera el código fuente y publica el SDK con la nueva versión. Esto sirve para que los desarrolladores puedan adaptar sus aplicaciones a dicha versión.

Suelen incluir un soporte para la detección de errores, un entorno de desarrollo integrado y otro tipo de utilidades. Incluyen también algún tipo de explicación de lo que contiene, códigos de ejemplos y un manual de uso. La mayoría de los kits de desarrollo de software son gratuitos y se distribuyen libremente por internet, para así fomentar la colaboración de los desarrolladores para mejorar su producto y para que utilicen su lenguaje. También se utilizan como herramienta de *marketing*, ya que por ejemplo, si animan a la gente a programar sus propias aplicaciones, comprarán productos para verlos, porque también les agrada seguir creando para ellos.

También existen kits de desarrollo de Java, DirectX y *frameworks*, en los que se basan por ejemplo los juegos para Windows o las aplicaciones que utilizan formularios.

3.5.3.2. Eclipse

Es una herramienta profesional al alcance de todos pese a que Eclipse está escrito en su mayor parte en Java (salvo el núcleo) y que su uso más popular sea como un IDE para Java, Eclipse es neutral y adaptable a cualquier tipo de lenguaje, por ejemplo C/C++, Cobol, C#, XML, entre otros. La característica clave de Eclipse es la extensibilidad. Eclipse es una gran estructura formada por un núcleo y muchos *plug-ins* que van conformando la

funcionalidad final. La forma en que los *plug-ins* interactúan es mediante interfaces o puntos de extensión; así, las nuevas aportaciones se integran sin dificultad ni conflictos.

Eclipse fue producto de una inversión de cuarenta millones de dólares de IBM en su desarrollo, antes de ofrecerlo como un producto de código abierto al consorcio Eclipse.org que estaba compuesto inicialmente por Borland e IBM. IBM sigue dirigiendo el desarrollo de Eclipse a través de su subsidiaria OTI (Object Technologies International), creadora de Eclipse. OTI fue adquirida por IBM en 1996 y se consolidó como gran empresa de desarrollo de herramientas orientadas a objeto desde la popularidad del lenguaje Smalltalk. OTI era la división de IBM en la que se generaron los productos Visual Age, que marcaron el estándar de las herramientas de desarrollo orientado a objetos.

Muchos conceptos pioneros en Smalltalk fueron aplicados en Java, creando *Visual Age for Java* (VA4J). VA4J fue escrito en Smalltalk. Eclipse es una reescritura de VA4J en Java. La base para Eclipse es la plataforma de cliente enriquecido (del inglés *Rich Client Platform RCP*). Los siguientes componentes constituyen la plataforma de cliente enriquecido:

- Plataforma principal, inicio de Eclipse
- Ejecución de *plug-ins* OSGi, una plataforma para integrar distribuciones.
- El *Standard Widget Toolkit* (SWT), un *widget toolkit portable*
- JFace, manejo de archivos, manejo de texto, editores de texto
- El *workbench* de Eclipse, vistas, editores, perspectivas y asistentes

El entorno de desarrollo Eclipse, incluyendo sus *plugins*, está desarrollado por completo en el lenguaje Java. Un problema habitual en herramientas Java (como NetBeans) es que son demasiado “pesadas”. Es decir, necesitan una

máquina muy potente para ejecutarse de forma satisfactoria. En gran medida, estas necesidades vienen determinadas por el uso del API Swing para su interfaz gráfico. Swing es una librería de *widgets* portable a cualquier plataforma que disponga de una máquina virtual Java, pero a costa de no aprovechar las capacidades nativas del sistema donde se ejecuta, lo cual supone una ejecución sensiblemente más lenta que la de las aplicaciones nativas.

SWT es una librería de *widgets* equivalente a *Swing* en la cual se aprovechan los *widgets* nativos del sistema sobre el que se ejecuta. El hecho de aprovechar los *widgets* nativos permite que la ejecución de interfaces de usuario sea mucho más rápida y fluida que si se utilizase Swing y, además, siempre dispone del “*Look and Feel*” del sistema, sin necesidad de “emularlo”.

La contrapartida es que la librería SWT es nativa, es decir, es necesario disponer de una librería SWT específica para cada sistema operativo. Existen versiones de SWT para los S.O. más habituales, incluyendo Windows, Linux, HP-UX, MacOS, etc.

3.5.3.2.1. Obtener, instalar y ejecutar Eclipse IDE

El IDE Eclipse se puede obtener bajándolo directamente del sitio web oficial del Proyecto Eclipse. www.eclipse.org, o de cualquier página de internet autorizada. Existen versiones instalables para cualquier plataforma que soporte la librería SWT, descargas que incluyen el código fuente y que incluyen los *plug-ins* más habituales. Además, de este mismo sitio, se puede descargar la librería SWT independientemente y su SDK. Como Eclipse está escrito en Java, es necesario para su ejecución, que exista un JRE (*Java Runtime Environment*) instalado previamente en el sistema. La instalación de Eclipse es tan sencilla,

como descomprimir el archivo descargado en el directorio que se estime conveniente.

3.5.3.3. APP Inventor

Si se está interesado en el mundo de la programación de aplicaciones móviles y se busca una herramienta que facilite la tarea, App Inventor es una de las mejores opciones. App Inventor es un entorno de desarrollo visual para Android originalmente propuesto por Google y finalmente implementado por el MIT (Massachusetts Institute of Technology) con el apoyo del gigante americano.

Su gran ventaja frente a otras plataformas similares es la potencia de su entorno de trabajo: se encontrarán muchas funciones ya definidas en forma de bloques y la flexibilidad a la hora de combinarlas permitirá potenciar al máximo la creatividad.

Su carácter gratuito la hace recomendable para todo tipo de usuarios. Si se quiere exprimir al máximo todas sus posibilidades, se necesita un cierto tiempo de adaptación y aprendizaje de la herramienta. Lo ideal es que se tengan unas mínimas nociones generales sobre programación para poder diseñar más fácilmente el flujo de acciones y comprender en profundidad la utilidad de cada bloque, aunque no será necesario que se modifique el código fuente.

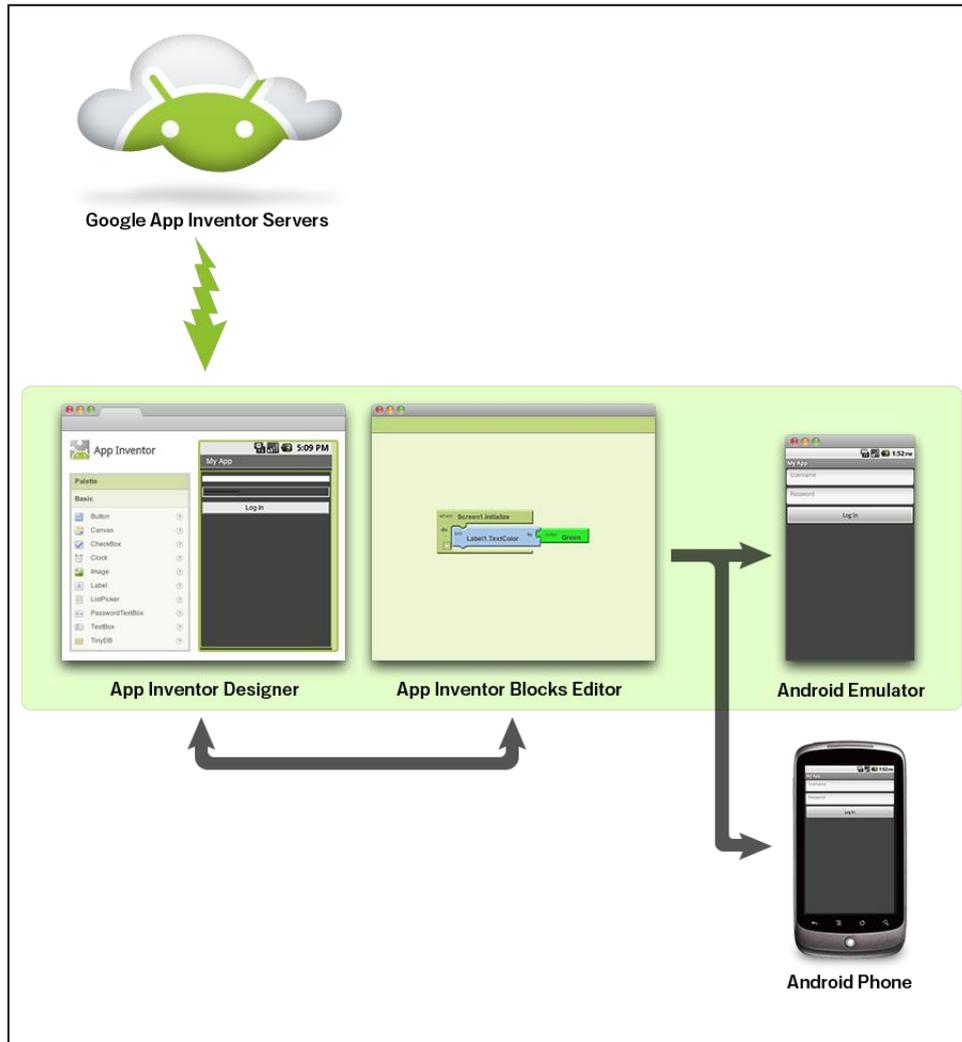
Para iniciar con el programa se tiene que ingresar a la web oficial de App Inventor. Se necesita tener una cuenta Google para acceder al servicio. App Inventor está compuesto por un conjunto de herramientas que abarcan las diferentes tareas en la construcción de la aplicación: diseño, definición del

comportamiento y simulación. Antes de pasar a utilizarlas se tiene que comprobar que se cuenta con todo lo requerido. Se debe ir a la página oficial y seleccionar *Setup*. Se le da clic en *Test your system* (prueba tu sistema) y esto chequea el navegador y el sistema.

Es necesario tener instalado Java y los sistemas operativos soportados son Mac OS X 10.5/10.6, Windows XP/Vista/7, Ubuntu 8+ y Debian 5+. Cuando se acabe, hacer clic en *done* (hecho). Luego se pasará a la descarga del software de escritorio App Inventor. En el último paso, escoger si se desea probar las aplicaciones con un teléfono o utilizar el emulador. Este último simulará un dispositivo Android que se podrá manejar a través del ordenador como si de un móvil real se tratara.

Se ejecuta vía web; es necesario ejecutar una pequeña aplicación para mostrar el editor de bloques (*block editor*) y el emulador, incluso las aplicaciones se guardarán en la web. Por lo cual se puede acceder a los proyectos desde cualquier equipo conectado a internet. Permite conectarse directamente al equipo Android, aunque en algunos casos no será posible; por ello incluye un emulador para probar las aplicaciones que se desarrollen. Permite guardar el proyecto en PC, para enviarlo a otros usuarios o para hacer copia de seguridad y también descargar la aplicación .apk compilada para ser instalada en el dispositivo Android. Por lo cual no se necesita ningún tipo de compilador ni nada por el estilo, todo se realiza en el entorno de App Inventor.

Figura 32. **Secuencia de programación con APP Inventor**



Fuente: *APP Inventor*. <http://appinventor.mit.edu/explore/sites/all/files/support/main/AppInventor-Doc-Diagram.png>. Consulta: 30 de abril de 2015.

3.5.3.3.1. **Panel de diseño**

El panel de diseño es accesible desde el navegador. En la página principal, se presiona en *Invent/Create Mobile Apps*. En *My Projects* se

selecciona *New* y se introduce un nombre para el nuevo proyecto. Una vez dentro, se pueden explorar todas sus opciones.

En la paleta Basic se encuentran utilidades básicas como botones, inserción de imágenes, cuadros de texto, etiquetas, lista de elementos, casilla para contraseñas y otras funciones más. La sección media contiene componentes como acceso a la cámara de fotos y vídeo, galería de fotografías, sonido y reproductor de vídeo. Social incluye los elementos necesarios para seleccionar contactos, números de teléfono de la agenda y correos electrónicos. Sensores, por su parte, brinda la oportunidad de utilizar sensores para obtener información, tal como la localización geográfica o la orientación del dispositivo. Para conocer algún detalle más sobre cada una de ellas, se le da clic en el símbolo de la interrogación que se encuentra junto a su título.

Figura 33. Panel de diseño de App Inventor



Fuente: *APP Inventor*. <http://beta.appinventor.mit.edu/learn/tutorials/quizme/QuizMeAssets/designer.png>. Consulta: 30 de abril de 2015.

El visor de la pantalla simula la apariencia visual que tendrá la aplicación en el móvil. Para añadir un elemento a la pantalla hay que arrastrarlo desde la paleta y soltarlo en el visor. Los elementos que no tengan visibilidad hay que arrastrarlos también al visor y automáticamente se desplazarán debajo de él bajo la ventana “*Non-visible components*”.

Se denominan elementos no visibles porque no se muestran físicamente en el panel de diseño, como por ejemplo los archivos de sonido. Se prueba a agregar un fichero a través de la opción *Source* en *Media/Sound*. El procedimiento para agregar estos elementos es el mismo que para aquellos visibles: arrástralo hacia el visor. Aparecerán listados en la parte inferior del panel, bajo el epígrafe *Non-visible components*. Otros elementos similares son cámara, *Camcorder* o *Player*, Elementos Bluetooth, al igual que todos los incluidos en la sección LEGO MINDSTORMS. Esta última incluye componentes, registrados bajo la marca Lego, que permitirán construir juegos de robótica.

Components muestra la lista de los componentes que se han colocado en el proyecto. Cualquier componente que haya sido arrastrado y soltado desde la paleta al visor aparecerá ahí. Si se quiere borrar alguno, esto se hace en la lista de componentes donde está el botón que permite borrarlo.

Cada vez que en el visor se seleccione un componente, en *Properties* aparecerán todos los detalles que se puedan cambiar de ese componente. Por ejemplo, al hacer clic sobre un componente *TextBox* se podrá cambiar en *Properties* su color, texto, fuente, entre otros. Para aquellos que hayan usado antes entornos de desarrollo del tipo de Visual Studio en Windows o Gambas en Linux, le será muy familiar esta forma de trabajar.

Este es el entorno en el que se diseña la aplicación. Para poder realizar la programación es necesario hacer clic en el botón de la esquina superior derecha con el texto “*Open the blocks editor*”. Eso hará que el navegador avise de que va a descargar una aplicación que arrancará con *Java Web Start*. Mientras se descarga la aplicación aparecerá en la parte inferior de la pantalla un texto indicativo de la carga.

3.5.3.3.2. Editor de bloques

Al terminar la descarga se generará un fichero Java con extensión JNLP que se deba guardar en la PC. Al abrirlo se accederá al editor de bloques. Esta ventana servirá para realizar el programa de la aplicación. En la pestaña *My Blocks* se verá la lista de todos los bloques añadidos en el panel de diseño.

Si se selecciona un botón, por ejemplo, se observan sus opciones, que aparecerán en forma de piezas de rompecabezas. Hay tipos de piezas que indican acciones y otras para componentes. La ventana editor de bloques está dividida en dos partes muy claras: un enorme lienzo en “blanco” a la derecha donde se irán arrastrando los bloques de programación y a la izquierda los menús que dan acceso a esos bloques.

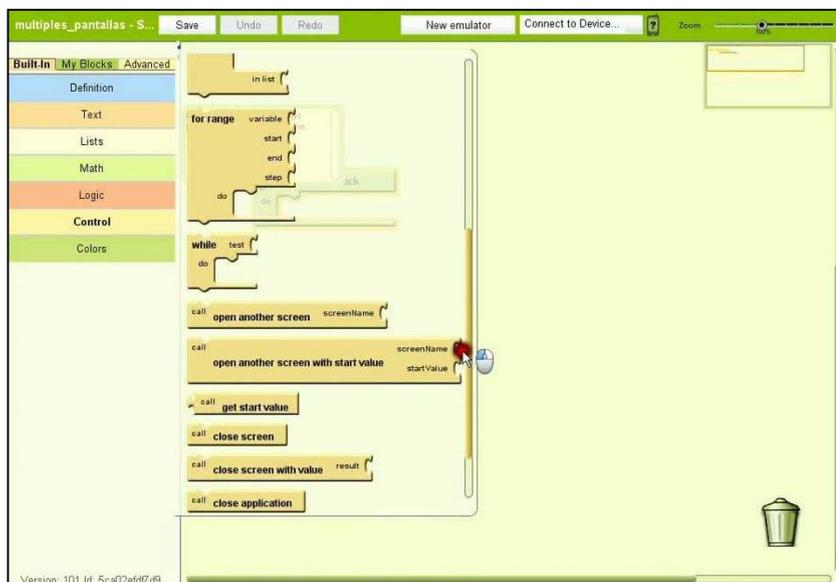
El menú está formado por tres pestañas: *Built-In*, *My Blocks* y *Advanced*, y bajo cada una de ellas aparecen a su vez distintos menús que permitirán seleccionar los bloques disponibles en cada caso.

- *Built-In* contiene los bloques que componen el lenguaje de programación propiamente dicho, distribuidos en las secciones *Definition*, *Text*, *Lists*, *Math*, *Logic*, *Control* y *Colors*.

- *My Blocks* ya se ha nombrado anteriormente y contendrá una lista de los elementos que hayan sido colocados en la pantalla *Viewer* del diseñador, para poder acceder mediante el programa a sus propiedades y eventos.
- *Advanced* es una pestaña que permite acceder de forma global a algunas propiedades comunes a todos los elementos que se han colocado en el Visor y que sean de un mismo tipo. Por ejemplo, al colocar un par de botones aparecerá “*Any Button*” y se podrán cambiar ahí algunas propiedades que afecten simultáneamente a los todos los botones.

En la parte superior derecha del editor de bloques de programación se pueden ver también los botones que dan acceso a conectar con un teléfono Android que estuviera enchufado al ordenador o lanzar el emulador.

Figura 34. **Editor de bloques de APP Inventor**



Fuente: *APP Inventor*. <http://appinventor.mit.edu/>. Consulta: 30 de abril de 2015.

3.5.3.3.3. Utiliza el emulador

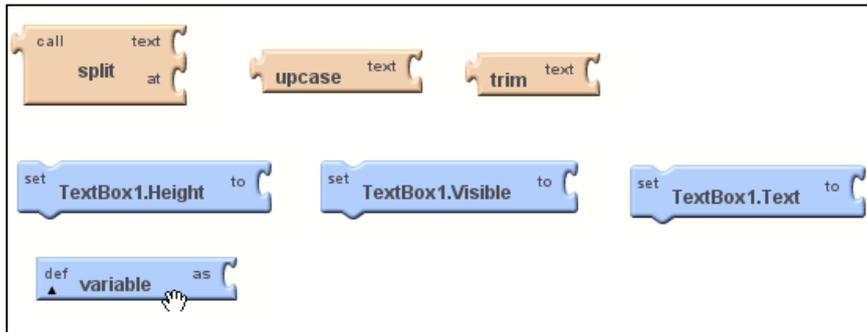
Se pueden desarrollar nuestras aplicaciones incluso si no se cuenta con un *smartphone* y se desea chequear su correcto funcionamiento. En el editor de bloques, al presionar en *New emulator* y se desplegará un teléfono Android típico. Para conectar el editor con el teléfono, se debe presionar en *Connect to Device* y seleccionar el emulador. La transferencia de información puede llegar a tardar dos o tres minutos. Después se accede al menú de aplicaciones como si de un dispositivo real se tratara.

El procedimiento para utilizar un móvil Android desde el editor es análogo. Se puede conectar al ordenador mediante un puerto USB y antes de iniciar la simulación, comprobar que la configuración está habilitada para hacer este tipo de uso del teléfono. Para ello, se deben seguir los pasos que se encuentran en la página oficial de la aplicación. Recordar que por ahora los archivos se encuentran exclusivamente en el ordenador.

3.5.3.3.4. Elementos de programación en App Inventor

- Sentencias: al programar se necesita a menudo decirle al ordenador que haga algo. En App Inventor existen numerosos bloques que son sentencias de programación. Se distinguen rápidamente porque expresan una acción a realizar con un verbo en imperativo. Los más habituales son *call*, *set* y *def* (abreviatura de define). Además, los bloques que son sentencias solo están disponibles en dos colores: azul para modificar variables o propiedades de un objeto y naranja para llamar a funciones.

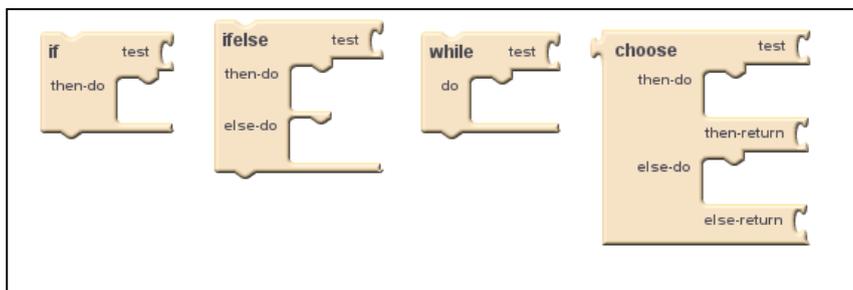
Figura 35. **Bloques de sentencias**



Fuente: *APP Inventor*. <http://appinventor.mit.edu/>. Consulta: 30 de abril de 2015.

- Condiciones: con frecuencia se necesita realizar distintas acciones en función de que ocurra o no algo. En App Inventor, para condicionar este programa, se tienen las estructuras *if-then*, *if-then-else*, *while* y *choose*. En todos estos bloques se puede ver cómo se tiene en la parte superior un espacio para encajar el *test* o pregunta que hace el programa para decidir por dónde continuar su flujo. Las tres primeras estructuras son habituales en la mayoría de los lenguajes, la última, *choose*, es prácticamente igual a *if-then-else* con la diferencia de que permite devolver una variable en función de que se cumpla o no el test.

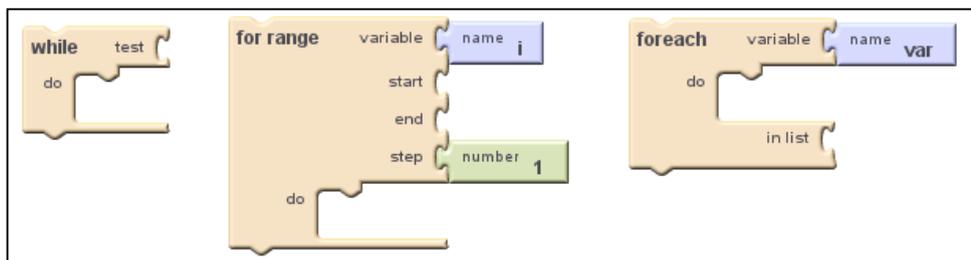
Figura 36. **Bloques de condiciones**



Fuente: *APP Inventor*. <http://appinventor.mit.edu/>. Consulta: 30 de abril de 2015.

- Bucles: un bucle provoca la ejecución repetida de varias sentencias. En App Inventor se dispone de los bucles *while*, *for range* y *foreach*. El primero ejecutará el bloque de sentencias que se encajen en el apartado mientras se cumpla la condición que refleje el *test*. *for range* es el típico bucle “*for*” de otros lenguajes de programación y, como tal, tiene las opciones de elegir el nombre de la variable que itera su comienzo, fin y el tamaño de los saltos de iteración. Finalmente, *foreach* es un iterador sobre los elementos de una lista.

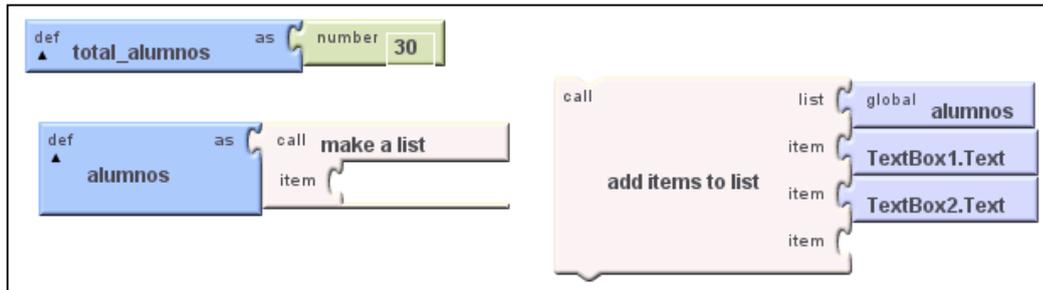
Figura 37. Bloques de bucles



Fuente: APP Inventor. <http://appinventor.mit.edu/>. Consulta: 30 de abril de 2015.

- Variables: se dispone de las herramientas para definir y cambiar variables. El tipo puede ser numérico o un texto y se declara al asignarle un valor por primera vez. En la figura 39 se ve cómo se declara una variable llamada “total_alumnos” y se le da un valor numérico de 30. Además se ven dos bloques muy importantes en App Inventor. Su lenguaje cuenta con una estructura de datos llamada *list* (lista) que es similar a los *Arrays* de otros lenguajes de programación. Se ve el bloque con el que se define la lista “alumnos” (se define vacía al no añadirle ningún *item*) y también se ve el bloque en el que se le añaden dos elementos a la lista alumnos. En este ejemplo se le añade el texto que contiene los cuadros de texto “*TextBox1*” y “*TextBox2*”.

Figura 38. Bloques de Variables



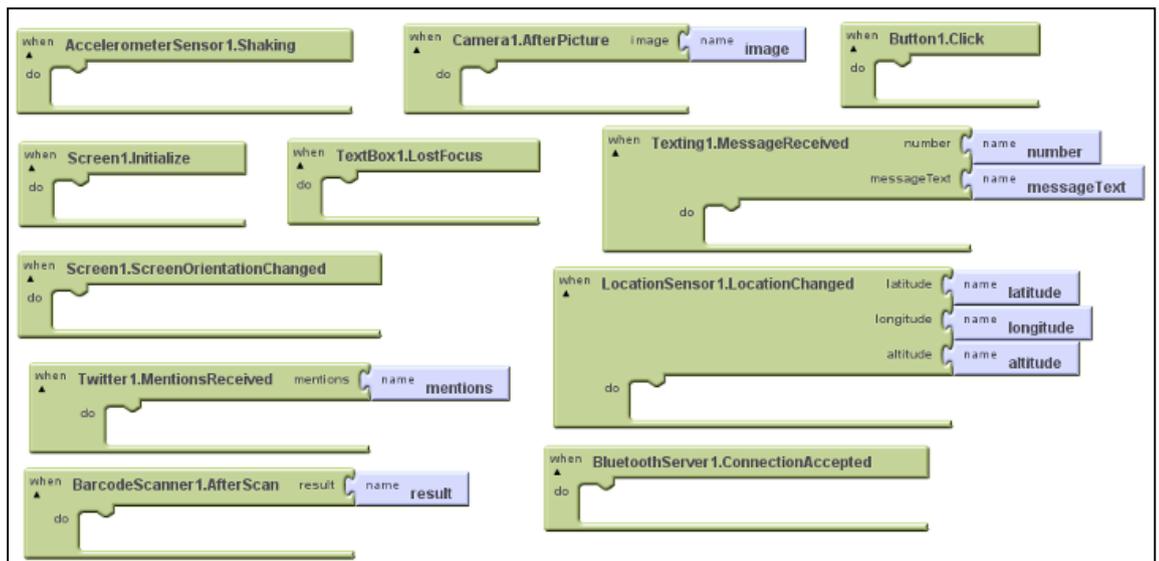
Fuente: *APP Inventor*. <http://appinventor.mit.edu/>. Consulta: 30 de abril de 2015.

- **Eventos:** es indispensable que los programas sean capaces de responder a la interacción del usuario con el interfaz. Es ahí donde toman su importancia los bloques de App Inventor, para ejecutar acciones, en respuesta a los eventos que se produzcan en el teléfono o el *tablet*. El nombre de los distintos eventos depende del objeto que los provoca. Un botón avisa cuándo se ha hecho clic sobre él, mientras que el acelerómetro avisa cuándo se ha agitado el móvil, la mensajería de cuándo se ha recibido un mensaje, el objeto de Twitter de cuándo han sido mencionados en esa red social, la cámara de fotos de cuándo se ha hecho una foto y así sucesivamente.

En la figura 40 se pueden ver algunos de estos eventos. De esta selección se aprecia que con App Inventor no solo se dispone de objetos que se pueden colocar en la pantalla como un botón o imagen, sino también de otros muchos objetos que dan acceso a las funcionalidades que tienen los teléfonos y tablets Android. Se tiene acceso a internet, Bluetooth, a hacer llamadas, enviar y recibir mensajes, las coordenadas de posición GPS, entre otros. También se puede apreciar cómo todos los eventos tienen la estructura “*When evento do*”, y son de color verde. Tienen este color

porque son parte de la sección “*My blocks*” en el entorno de desarrollo. “*My blocks*” es donde se leen o manipulan las propiedades de los distintos elementos que hayan sido colocados manualmente en el entorno de diseño: botones, brújula, etiquetas, cuadros de texto, entre otros.

Figura 39. **Bloques de eventos**



Fuente: *APP Inventor*. <http://appinventor.mit.edu/>. Consulta: 30 de abril de 2015.

4. DISEÑO DEL SISTEMA DE AUTOMATIZACIÓN

Para el diseño de todos los elementos que van a formar el sistema de automatización se realizará el estudio de las partes que van a ser controladas por el sistema en conjunto con el usuario.

4.1. Elementos que se van controlar en la casa

Tomando de referencia el plano de la casa de la figura 6 (capítulo 1), se indicarán los elementos a controlar en la casa: iluminación, apertura de puertas y portón, control de persianas en las ventanas, regado de jardín y control perimetral de la casa.

4.1.1. Iluminación

El encendido y apagado de las luces de la casa se controlará mediante la aplicación del teléfono. Se contará con el control para 8 luces de cualquier tipo. Se puede tener el control de iluminación en cualquier ambiente de la casa. El control puede ser manual o una secuencia activada por el usuario donde se le indique el orden para encender o apagar luces. Para este proyecto los ambientes a controlar son los siguientes:

- Parqueo
- Sala
- Dormitorio 1
- Dormitorio 2
- Dormitorio 3

- Baño
- Cocina
- Patio

4.1.2. Puertas y portón eléctrico

La apertura y cerrado, de puertas y portón de la casa se hace activando cerraduras o relés de estado sólido, en el caso del motor del portón. Para el proyecto se controlará puerta principal, puerta del patio y el portón eléctrico.

4.1.3. Apertura de persianas de las ventanas

La luminosidad de los ambientes de la casa se puede controlar con persianas con motor eléctrico; para este caso, se controlarán las persianas de la sala y del dormitorio 1.

4.1.4. Regado del jardín

El regado del jardín va tener un control electrohidráulico que va a controlar el sistema en conjunto con el usuario, en el cual se va a temporizar el regado o planificar los tiempos de regado de cada día.

4.1.5. Control perimetral de la casa

Las puertas, ventanas, parqueo, patio y el portón de la casa podrán ser vigilados desde la aplicación del teléfono, indicando una alarma cuando en alguna de las zonas exista un movimiento. Las zonas de la casa a controlar son las siguientes:

- Ventana de la sala
- Ventana de dormitorio 1
- Ventana de dormitorio 2
- Puerta principal
- Puerta trasera
- Portón
- Patio
- Parqueo

4.2. Diseño del control

El control del sistema va a estar formado por varios circuitos; entre estos están: mando, potencia y comunicación. Lo primero que se debe de hacer es la conexión de los sensores y los actuadores que realizarán alguna tarea en la casa.

4.2.1. Conexión de los actuadores

Los actuadores con que va a contar la casa, van a ser las lámparas, motores, cerraduras de puerta y electroválvulas. Los actuadores son de 120VAC y 24 VDC.

4.2.1.1. Lámparas

La iluminación de la casa va ser a través de luces de cualquier tipo, incandescentes, de alógeno, de neón o de led. Para lo mismo se necesita instalar una lámpara normal. Después de haber determinado el tipo de lámpara que debe de ser establecida en el espacio, se procede a localizar una serie de herramientas: destornilladores, taladro, escalera, pinzas y por último, una

regleta de borne. Después de tener todos los materiales necesarios para el trabajo, se procede a eliminar la fuerza del interruptor, es decir, se corta el circuito eléctrico que va con el lugar que se trabaja. Se procede a inspeccionar la forma de conexión de la lámpara. Para este caso la conexión eléctrica solo va a tener los dos cables de línea y referencia que van a ir hacia el control de potencia.

4.2.1.2. Cerraduras de puerta

Las puertas utilizarán cerraduras electromagnéticas, las cuales presentan un fuerte nivel de seguridad, y la facilidad de manejo e instalación. Las cerraduras electromagnéticas cuentan con dos principales piezas, por un lado el electroimán, y por el otro una lámina metálica llamada pieza móvil o polar. El electroimán se coloca en el marco de la puerta, trabaja como imán en la medida que circule corriente por su bobina y cierra la puerta; al dejar de recibir corriente eléctrica permite la apertura de la puerta.

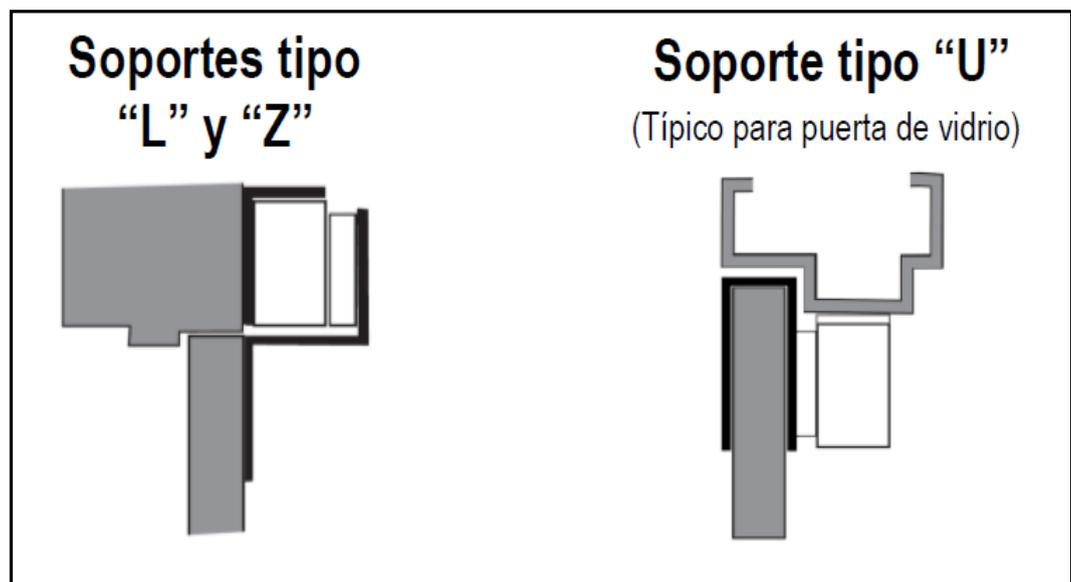
Todas estas cerraduras electromagnéticas son de tipo “*Failsafe*”, lo que significa que se mantienen cerradas solo mientras exista corriente eléctrica, a diferencia de los otros tipos de cerraduras eléctricas que funcionan del modo “*Failsecure*”, las cuales funcionan de modo contrario, cuando no hay electricidad se mantienen cerradas; por lo que las cerraduras electromagnéticas necesitan fuentes de energía de respaldo como UPS o baterías, para que cuenten con el nivel de seguridad alto.

La instalación de las cerraduras electromagnéticas es extremadamente sencilla, como regla básica el electroimán debe estar perfectamente alineado con la placa metálica (pieza móvil o pieza polar); para lograr esta alineación existen varios tipos de bases para estas cerraduras. Entre el tipo de bases se

encuentran las siguientes: base plana incluida con la cerradura, y este otro tipo de bases, las cuales se deben adquirir por separado: base tipo U, base tipo L, base tipo Z.

La mejor posición para la instalación de la cerradura electromagnética es en la parte interna de la puerta que se asegurará con el alambrado empotrado en el marco, para prevenir la manipulación de la unidad. La instalación eléctrica de la cerradura requiere dos alambres que llevarán la corriente eléctrica al electroimán.

Figura 40. **Soportes de instalación para cerraduras electromagnéticas**



Fuente: *Tecnología Electrónica SRL*. www.rdns.com.ar. Consulta: 5 de mayo de 2015.

4.2.1.3. Motor de portón

El portón que se instalará será uno corredizo, el cual utiliza un motor monofásico de 120/240 VAC. Antes de instalar el sistema del motor, es muy importante verificar las condiciones de funcionamiento del portón. Para eso se deben tomar las siguientes instrucciones:

- Mover la hoja del portón y observar si la misma corre libremente. El desplazamiento debe ser suave; esto quiere decir que el portón está listo para ser automatizado.
- Asegurarse también que las roldanas estén libres y el riel no esté sucio o torcido.
- En el caso de que esas condiciones no estén dadas, entonces el portón deberá pasar por una revisión y mejorar su condición de funcionamiento, para así ser automatizado.
- El portón deberá tener una estructura resistente y estar totalmente apoyado en el piso sobre roldanas; también en la parte superior a través de guías, en el caso de ser necesario.
- Las roldanas deberán tener el diámetro según las dimensiones del portón, con perfectas condiciones de funcionamiento y armadas de forma que el portón tenga estabilidad en todo su desplazamiento.

Teniendo en cuenta todos estos aspectos se instalará el sistema de portón corredizo, donde lo más importante es el motor y las cremalleras que servirán para que el motor abra o cierre una hoja del portón. Se necesitará realizar una plataforma de concreto dónde fijar la base de montaje. Calcular la altura más conveniente de la plataforma, para la fijación posterior de la cremallera en la hoja del portón.

Fijar la cremallera, teniendo cuidado en dejar la menor luz posible con el engranaje, pero permitiendo que se mueva libremente. Si la cremallera se instala muy ajustada contra el engranaje, provocará un desgaste rápido en el equipo, en caso contrario podrá provocar la quebradura de dientes. Permitir que estando el portón totalmente cerrado o abierto, queden sobrando algunos dientes de la cremallera en relación con el engranaje del accionador.

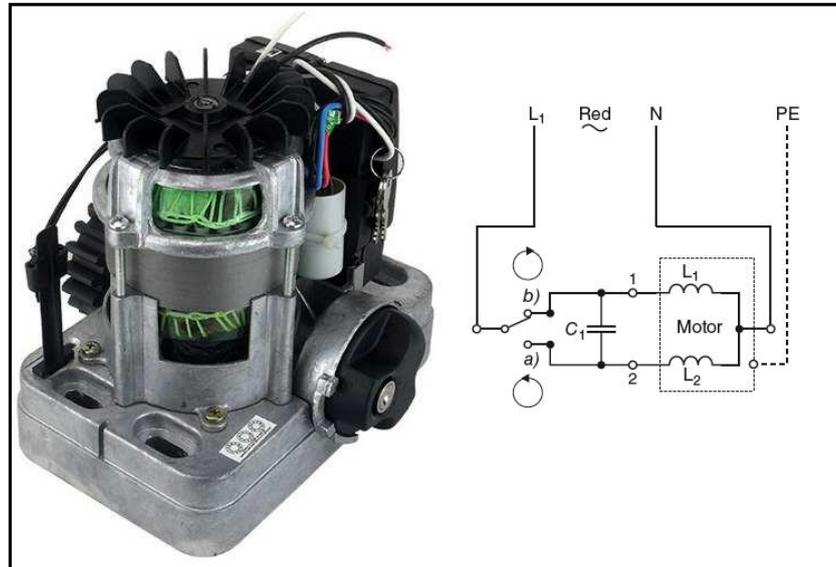
Montar la base del motor a la base de concreto. Verificar que la base esté adecuadamente nivelada. Utilizando ambas tuercas fijar la base del motor y colocar la tapa cobertora. Chequear el motor y asegurarse de que esté alineado con el portón. Fijar el motor al sistema de engranaje y cremallera.

La instalación eléctrica se realiza de acuerdo con el diagrama de conexiones que acompaña a cada modelo de motor. En varios casos se trata de motores monofásicos que utilizan un capacitor y un embobinado auxiliar de arranque. En los motores actuales, las bobinas de arranque se conectan con la red a través de un condensador en serie que, a la frecuencia de la red y la velocidad nominal del motor, produce un desfase tal entre las corrientes de los devanados de arranque y servicio, que se hace innecesario desconectarlas, por lo que estos motores ya no necesitan incorporar el interruptor centrífugo simplificando su constitución y funcionamiento.

Por seguridad, es recomendable instalar finales de carrera en ambos extremos del portón, para bloquear el paso de corriente hacia el motor. Para prevenir el descarrilamiento del mismo.

La mayoría de los motores pueden ser desbloqueados en caso de corte de luz, simplemente accionando una manilla o utilizando una llave. Esto permite que el portón pueda ser abierto manualmente.

Figura 41. **Motor monofásico de portón corredizo y conexión interna**



Fuente: *Starligh S.A.* <http://www.starligh.com/automatismos/dz-450.html>. Consulta: 5 de mayo de 2015.

4.2.1.4. **Motores en persianas de ventanas**

Se requiere tener el control de subir y bajar las cortinas o persianas de las ventanas para controlar la luminosidad de una habitación. Se pueden colocar sistemas motorizados, estos son fáciles de instalar, solo se debe contar con el kit completo. El kit para persiana incluye el motor y las bases para colocarlos en la ventana.

Dependiendo del diámetro del eje que se tenga en la persiana será necesario utilizar un motor más sencillo o más grande. Lo primero que se hace es colocar el motor en el eje de las cortinas y ubicar los soportes en la pared donde va a ir ubicada la cortina. Una vez se tengan los soportes listos, se coloca el eje con el motor sobre ellos y se comprueba que quede bien

asentado. Colocar el cable bien para que no se pueda enganchar al girar el eje y se prepara una regleta dónde enchufar a 120VAC los cables, para ajustar el motor antes de hacer el cableado definitivo.

En el presente caso se conectará el cable azul (neutro) y marrón o negro, dependiendo del sentido de giro. Se coloca la cortina o persiana de la ventana y se ajustan los finales de carrera del motor, que indican cuándo la cortina está arriba o abajo. Estos finales de carrera le bloquean el paso de corriente al motor para que no siga girando, en cualquier sentido.

Es bueno hacer un par de recorridos completos para comprobar que los fines de carrera están correctamente ajustados, pero procurando tampoco darle mucho trabajo al motor porque se calienta rápido.

Figura 42. **Instalación de persiana motorizada y ajuste de finales de carrera del motor**



Fuente: *Domotica para todos*. <http://domoticaparatodos.com/2014/06/guia-persianas-motorizadas/>. Consulta: 5 de mayo de 2015.

4.2.1.5. Electroválvula hidráulica

Para el riego de jardín se utilizan válvulas eléctricas del tipo normalmente cerradas, de forma que cuando el control da las órdenes de riego, la válvula automática se pone en funcionamiento, abriéndose y dejando pasar el caudal para alimentar los aparatos de riego que componen un sector. Una vez acabado el tiempo de riego, la válvula deja de funcionar y se cierra, paralizando el sector.

El primer paso para diseñar un sistema de riego residencial es medir la propiedad e indicar la ubicación de las electroválvulas y la tubería. En una hoja de papel puede hacerse un esquema de la casa y colocar las medidas en dicho esquema. Hay que decidir en qué lugar se instalarán los aspersores y difusores. Los aspersores de mediano alcance deberán colocarse a una distancia entre 5 y 8 metros.

No se debe enterrar la electroválvula directamente. Instalar una caja de registro para lograr acceso fácil a la electroválvula. Utilizar el cable para conectar la electroválvula con el control. Para la instalación del cable, colocarlo en la zanja desde el control hasta la electroválvula. Es mejor proteger el cable de futuras excavaciones instalándolo directamente por debajo del tubo, cuando sea posible. Dejar una vuelta de cable para expansión en cada cambio de dirección. El lazo asegurará que los cables no se instalarán demasiado apretados y reducirán la posibilidad de estiramiento. Conectar los cables a la electroválvula con los conectores impermeables. No es aconsejable realizar empalmes, excepto en las válvulas automáticas, debiendo realizarse estos con cinta vulcanizable.

Figura 43. **Electroválvula para riego**



Fuente: *Saneamientos Barrios*. <http://www.saneamientos-barrios.com/pdf/electrovalvula-2500.pdf>. Consulta: 5 de mayo de 2015.

4.2.2. Conexión de sensores

Una casa puede estar protegida por un sistema de seguridad que utiliza sensores. Los sensores se conectan y accionan una señal a la alarma, si la puerta se abre o se retira de sus bisagras o si hay movimiento en las áreas de afuera de la casa, para indicarle a las personas algún tipo de alarma. La instalación de los sensores es un proceso sencillo.

Para el sistema de control perimetral de la casa se necesitará instalar sensores de puerta, ventana y sensores de movimiento para el patio y parqueo.

4.2.2.1. Sensores magnéticos

Los dispositivos utilizados son los sensores magnéticos. Hay una buena cantidad de modelos de sensores magnéticos en el mercado, pero todos

consisten en dos piezas apareadas: el imán y el contacto de ampolla. Eléctricamente, es un interruptor momentáneo, tal como un pulsador, y se presenta con contacto normalmente abierto (NA), normalmente cerrado (NC) o inversor (NA y NC).

Su funcionamiento es sencillo, los contactos de la ampolla están imantados, y en presencia del campo magnético del imán cambian de posición abriendo o cerrando un circuito. No es necesario que ambas piezas se toquen; bastará que la ampolla esté dentro del campo magnético del imán. La ventaja de los detectores magnéticos sobre otros sensores es que no tienen contacto físico con las partes móviles de la abertura, ya que su contacto está dentro de una ampolla de vidrio al vacío, lo que lo hace más duradero y eficaz al no depender de rugosidades o bordes del marco de la puerta.

Para la instalación de los sensores se debe saber que como el contacto magnético debe estar dentro del campo generado por el imán, la posición entre ambas partes requiere cierto cuidado. En principio, la distancia entre el imán y el contacto debe ser la menor posible dentro de la especificada por el fabricante. Este valor se conoce como brecha o *gap*. Cuanto mayor sea el valor especificado, es más sencillo regular el magnético para un buen funcionamiento. Las aberturas que tengan juego o que no encajen correctamente presentan inconvenientes para estos dispositivos. Los sensores magnéticos comunes deben estar paralelos a la abertura y entre sí, además de alineados y no desplazados.

Por lo general, conviene atornillar el sensor magnético, ya que le confiere gran durabilidad; de ser necesario, adherirlo; se debe usar un adhesivo epoxi ultra fuerte o adhesivo de doble faz especial para este uso. No es óptimo en todos los casos, con el tiempo se puede desprender el sensor magnético

generando una falsa alarma. No se recomienda pegamento instantáneo por su rigidez. Es preferible un adhesivo elástico.

Figura 44. **Instalación de sensor magnético en puerta**



Fuente: *Tecno seguro*. http://www.tecnoseguro.com/images/stories/General/Magnetico_en_Puerta.jpg. Consulta: 5 de mayo de 2015.

4.2.2.2. Sensor de movimiento

Un sensor de movimiento es utilizado comúnmente para encender una luz, cuando una persona se mueve frente a él; de esta manera esa tarea queda automatizada, y además en la mayoría de los casos sirve como método de seguridad, pues si el sensor está ubicado en la entrada de la casa, y el que entra es un intruso, activará una alarma.

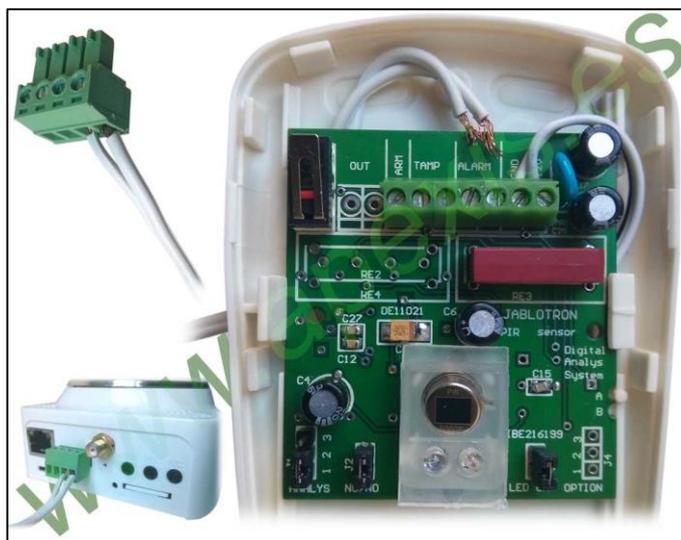
Para instalar un sensor de movimiento elegir un lugar adecuado para montar el sensor por infrarrojos. Identificar un lugar donde el sensor no dará lugar a una falsa alarma; este detectará el calor de los conductos de ventilación y el de las ventanas o las luces exteriores que brillan a través de una ventana. Se debe consultar el manual de instrucciones del sensor y seguir las recomendaciones del fabricante respecto de qué tan lejos debe estar de

cualquier fuente de calor y cómo apuntar el lente de la unidad de detección de movimiento.

Los cables en su interior normalmente respetan el código de colores (marrón-fase; azul-neutro). Entre sus principales características están las de alcance y ángulo (grados) de detección de movimiento, nivel de luminosidad a la que actúan, tiempo de cierre del circuito y la carga que controlan (potencia). La mayoría de las veces constan de dos reguladores o selectores, uno de tiempo de duración de encendido de la lámpara, el cual va de los segundos hasta los minutos y otro del nivel de luz a la que actuarán tres luxes o más.

Los sensores cuentan con señales de un relé que indican cuando hay presencia de alguien o detectan algún movimiento. Cuenta con un contacto N.O, N.C. y el común. Estas son las señales que van a llevarse al control.

Figura 45. **Conexión de un sensor de movimiento**



Fuente: *Apexis Electronic*. http://www.apexis.es/images/apexis_web. Consulta: 5 de mayo de 2015.

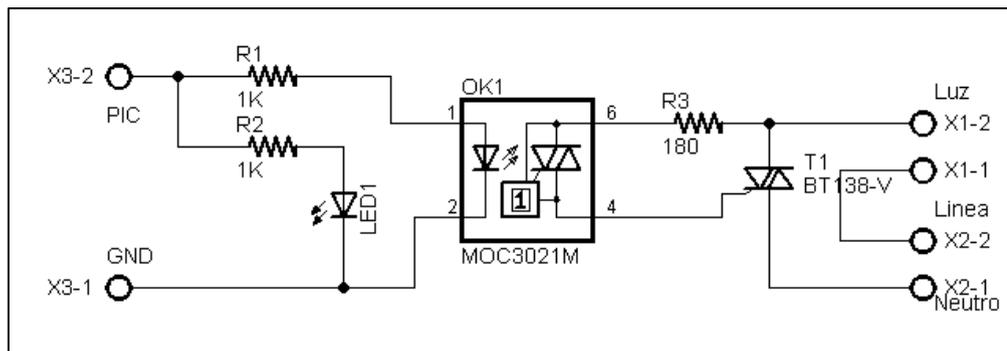
4.2.3. Circuitos de potencia

Para los circuitos de potencia se van a utilizar semiconductores y transistores, en el caso de las cerraduras magnéticas y la electroválvula y triac en las luces, motor de portón y motores de persianas.

4.2.3.1. Circuitos de iluminación

Para el circuito que va controlar la potencia de las luces se utiliza el diagrama de la figura 46, el optoacoplador se encarga de separar impedancias, para manejar voltaje DC de un lado del circuito y voltaje AC del otro. El led indica que el mando principal le envía señal para que encienda la luz. El triac se encarga de manejar la potencia de la luz. El triac es de 4 amperios 240VAC máximo, con lo que se pueden manejar luces de alta potencia. En la bornera marcada con X2-2 se conecta la línea, en X2-2 la referencia, del voltaje alterno a utilizar. En la bornera X1-1 y X1-2 se conecta la lámpara.

Figura 46. Control de potencia de luces



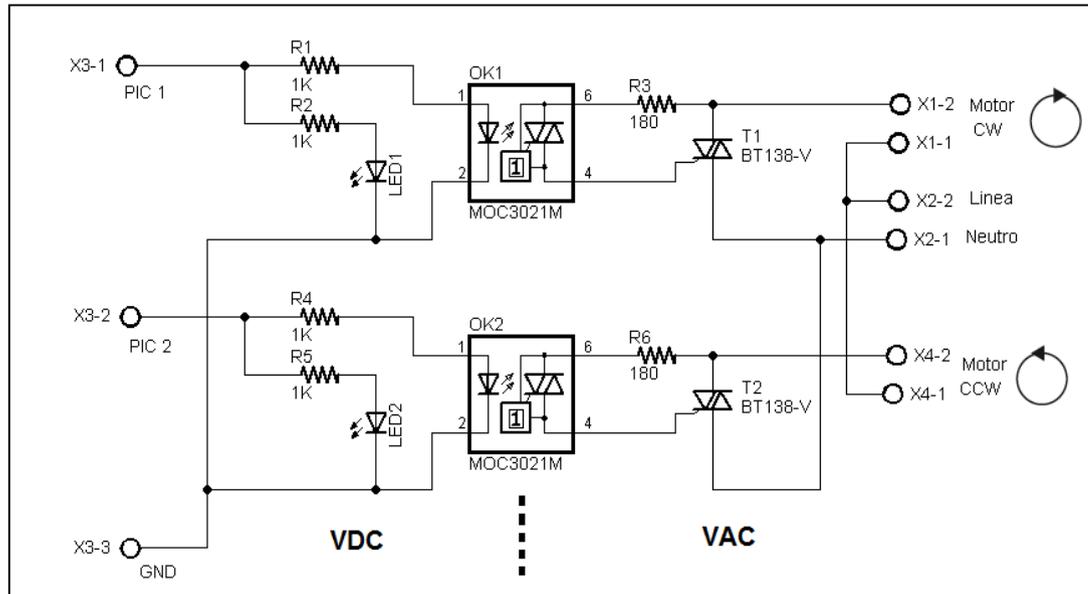
Fuente: elaboración propia, empleando Eagle 6.2.0.

4.2.3.2. Circuitos de motor alterno

Para manejar la potencia de los motores que están en el sistema, se utilizará el diagrama de la figura 47. Este circuito utiliza optoacopladores que separan las impedancias para manejar corriente DC en un lado y corriente AC del otro. Del lado izquierdo del circuito se trabaja con corriente directa; en la bornera X-3-1 y X-3-2 se conectan salidas del PIC que le mandan la señal para que encienda el motor; los led 1 y 2 indican qué puerto está habilitado. Del lado de la derecha se trabaja con corriente alterna. El elemento semiconductor de potencia es el triac BT138 que puede manejar una corriente de 8A a 240 VAC, lo que permite controlar motores de alta potencia.

El motor monofásico que se va a utilizar es del tipo que se muestra en la figura 41; el cual para controlar su giro utiliza diferentes embobinados. Cada embobinado del motor lo controla un triac, en la bornera X1-1 y X1-2; se debe conectar el embobinado del motor para el giro en sentido horario del motor y en la bornera X4-1 y X4-2 el embobinado del giro en sentido anti horario del motor. En la bornera X2-2 se conecta la línea de voltaje alterno y en X2-1 la referencia del voltaje. Este circuito se utiliza en el motor del portón y en el motor de la persiana de ventana.

Figura 47. **Control de potencia de motor AC**



Fuente: elaboración propia, empleando Eagle 6.2.0.

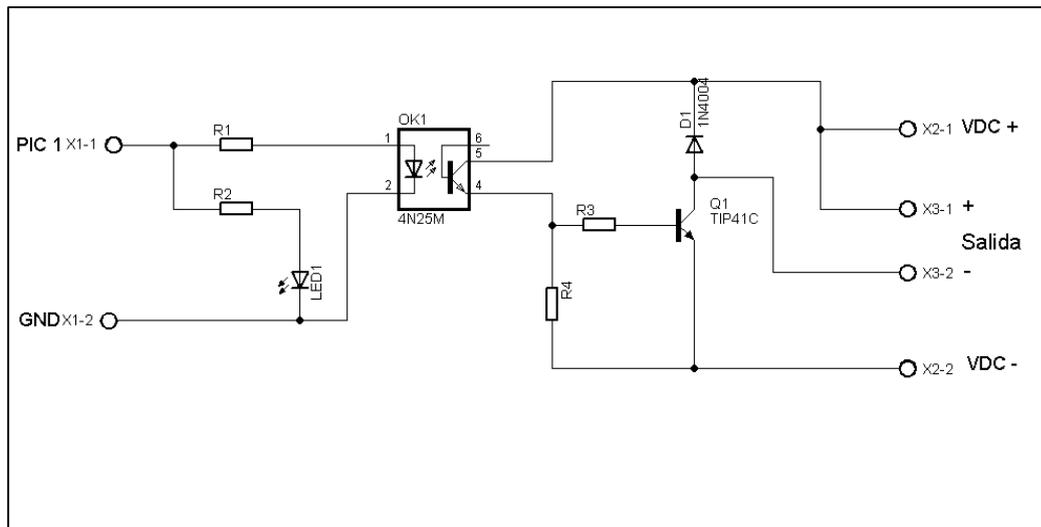
4.2.3.3. **Circuito de potencia para cerraduras y electroválvula**

Para el manejo de las cerraduras magnéticas y electroválvulas se utilizará el circuito de la figura 48. Utiliza un optotransistor que realiza el desacople de impedancias para utilizar una distinta fuente de alimentación para los actuadores de las puertas y del sistema de riego. Esto evita que el microcontrolador reciba ruido electromagnético, debido a que las bobinas cuando se activan o desactivan tienden a generar picos de corriente que provocan mal funcionamiento en el control.

El elemento que controla la potencia de los actuadores es el transistor bipolar TIP41C que puede manejar una corriente de 6A max. La conexión del circuito en la parte izquierda se hace directamente al microcontrolador, en la

bornera X1-1 se conecta la terminal del PIC y en la bornera X1-2 el GND. El led 1 indica cuando existe una señal del PIC. Del lado de la potencia en la bornera X2-1 y X2-2, se conecta la fuente de alimentación en DC que utilice el actuador. Para las cerraduras electromagnéticas y la electroválvula, se utilizan 24VDC. En la bornera X3-1 y X3-2 se conecta el actuador. Como se trata del voltaje directo, se toman en cuenta las polaridades de cada conexión.

Figura 48. **Control de potencia para cerradura magnética y electroválvula**



Fuente: elaboración propia, empleando Eagle 6.2.0.

4.2.4. Circuito de comunicación

Para realizar la comunicación entre el microcontrolador y el dispositivo Android, se utilizará como interfaz humano-máquina, vía Bluetooth, por medio de una tarjeta de comunicación Bluetooth Stick, de la empresa Mikroelektronika. El Bluetooth Stick es una tarjeta adicional que permite al microcontrolador poder comunicarse con los dispositivos inalámbricos.

La comunicación con un microcontrolador se realiza vía comunicación UART serie. Para conectar esta placa adicional a un microcontrolador se pueden utilizar seis terminales que se muestran en la figura 49. El voltaje de alimentación que utiliza esta tarjeta es de 3.3 VDC que se conectan en las terminales marcadas como 3.3v y GND.

Las terminales TX (transmisor) y RX (receptor) se conectan al microcontrolador. Las terminales CTS y RTS se utilizan para mandar señales que le indican al microcontrolador que está listo para recibir o transmitir datos. Tiene un rango de 100 metros de alcance y utiliza una frecuencia de 2402 - 2480MHz.

Figura 49. **Tarjeta Bluetooth Stick de la empresa Mikroelektronika**



Fuente: MikroElektronika. *Bluetooth Stick Manual*. p. 2.

El módulo debe configurarse utilizando los *Jumpers* que trae esta tarjeta. En la figura 49 se pueden observar los *jumpers* que están marcados con la letra J; dichos *jumpers* tienen las siguientes funciones:

- J1: permite la conexión entre RTS y el módulo Bluetooth RN 41;
- J2: permite la conexión entre CTS y el módulo Bluetooth RN 41;
- J3: conecta VCC-3.3V (*pull-up*) al pin PIO3 módulo Bluetooth, para habilitar el descubrimiento automático;
- J4: conecta VCC-3.3V (*pull-up*) al pin PIO4 módulo Bluetooth, para establecer los valores predeterminados de fábrica;
- J6: conecta VCC-3.3V (*pull-up*) al pin PIO6 en el módulo Bluetooth, para establecer como maestro Set BT (*HIGH* = modo *automaster*)
- J7: Conecta VCC-3.3V (*pull-up*) al pin PIO7 en el módulo Bluetooth, para establecer la velocidad de transmisión de 9600 bps.

Es recomendable restablecer los valores de fábrica; para esto se utiliza el *jumper* 4 realizando la siguiente secuencia: GPIO4 debe ser alta en el arranque y luego alternar baja, alta, baja, alta con un segundo de espera 1, entre las transiciones. Luego de esto el módulo retoma los valores de fábrica. Para mostrar los estados del módulo, la tarjeta cuenta con la terminal GPIO 5 que sirve para conducir un led; este parpadea a varias velocidades para indicar el estado. Los mensajes dependen de la velocidad de parpadeo que a continuación se enlistan:

- Parpadea a 1HZ, el módulo se puede detectar y está a la espera de una conexión.
- Parpadea a 10HZ, el módulo está en el modo de comandos.
- Se queda iluminado, el módulo está conectado a otro dispositivo a través de Bluetooth.

4.2.5. Circuitos de mando

El cerebro de todo el mando va ser el microcontrolador PIC16f877A, de este se disponen 33 terminales de entrada y salida. Un puerto de comunicación serial. Sus 33 terminales están divididas en 5 puertos. El puerto A 5 terminales E/S, el puerto B, C, D con 8 terminales de E/S y el puerto E con 3 terminales de E/S; en la tabla 3 se enumeran los puertos y la función que este va a hacer en el control.

Tabla IV. Puertos y funciones del microcontrolador

| Puerto | Número terminal | Función E/S | Función a controlar |
|---------------|------------------------|--------------------|--|
| RA0 | 2 | Entrada | Sensor de puerta principal |
| RA1 | 3 | Entrada | Sensor de puerta patio |
| RA2 | 4 | Entrada | Sensor de portón |
| RA3 | 5 | Entrada | Sensor de ventana sala |
| RA4 | 6 | Entrada | Sensor de ventana dormitorio P |
| RA5 | 7 | Entrada | Sensor de ventana dormitorio 1 |
| RE0 | 8 | Entrada | Sensor de movimiento parqueo |
| RE1 | 9 | Entrada | Sensor de movimiento patio |
| RB0 | 33 | Salida | Abrir portón |
| RB1 | 34 | Salida | Cerrar portón |
| RB2 | 35 | Salida | Subir persiana de ventana de la sala |
| RB3 | 36 | Salida | Bajar persiana de ventana de la sala |
| RB4 | 37 | Salida | Subir persiana de ventana del dormitorio principal |
| RB5 | 38 | Salida | Bajar persiana de ventana del dormitorio principal |
| RB6 | 39 | Salida | Puerta principal |
| RB7 | 40 | Salida | Puerta patio |

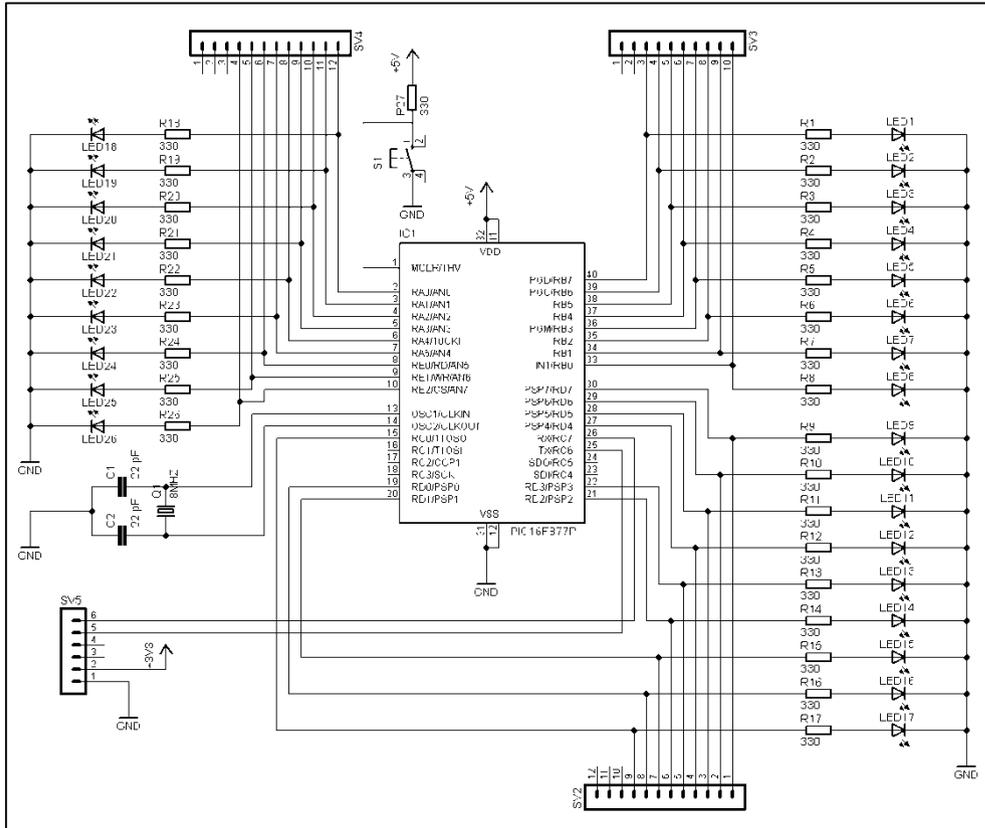
Continuación de la tabla IV.

| Puerto | Número terminal | Función E/S | Función a controlar |
|---------------|------------------------|--------------------|-----------------------------------|
| RD0 | 19 | Salida | Lámpara de dormitorio principal |
| RD1 | 20 | Salida | Lámpara de dormitorio 1 |
| RD2 | 21 | Salida | Lámpara de dormitorio 2 |
| RD3 | 22 | Salida | Lámpara de sala |
| RD4 | 27 | Salida | Lámpara de cocina |
| RD5 | 28 | Salida | Lámpara de baño |
| RD6 | 29 | Salida | Lámpara de patio |
| RD7 | 30 | Salida | Lámpara de parqueo |
| RC0 | 15 | Salida | Regado de jardín |
| RC1 | 16 | Salida | Conexión Bluetooth exitosa |
| RC6 | 25 | Salida | Transmisor de comunicación serial |
| RC7 | 26 | Entrada | Receptor de comunicación serial |

Fuente: elaboración propia.

En el circuito de la figura 51 se muestra la conexión del microcontrolador. En el puerto A y E se conectan los sensores de las puertas, ventanas y los sensores de movimiento; esto se hace en el conector SV4, los leds 18 al 26 indican el estado de los sensores. En los puertos B, D y C se conectan las salidas para los circuitos de potencia de los actuadores; para esto se utiliza el conector SV2 y SV3, los leds 1 al 17 indican qué actuador está activado. El pulsador S1 funciona como *reset* del microcontrolador. El oscilador del microcontrolador es de 8 MHZ. El conector SV5 se utiliza para mandarle voltaje y señales al módulo Bluetooth.

Figura 51. Circuito de mando principal



Fuente: elaboración propia, empleando Eagle 6.2.0.

4.2.5.1. Programa del microcontrolador

El programa del microcontrolador que realiza las tareas del sistema está hecho en el lenguaje Basic. Se realizó utilizando el software de programación Mikrobasic. En la figura se muestra el programa del microcontrolador. Se declaran las variables que se utilizarán para guardar la información que se recibe o se envía hacia el teléfono. Se configuran los puertos que se utilizarán como entradas y como salida. Luego se parametrizan las variables y los puertos para que inicien con un valor ya fijo.

Se inicia la comunicación serial, a una velocidad de 9600 BPS; luego el proceso donde se va estar recibiendo los *bytes* que envía el teléfono; estos datos los recibe el módulo Bluetooth y se los comunica al microcontrolador; esta información se guarda en la variable *i*. Dependiendo del valor de la variable *i*, se realiza una tarea definida en el menú selector. Si el teléfono le envía el dato 67, el microcontrolador le envía un *byte* con la información de los sensores.

Tabla V. Programa del microcontrolador

| | |
|--|---|
| <pre> program Casa_Automatizada dim i as byte dim dato as byte dim numero as byte dim key as byte TRISB = 0 TRISA = 255 TRISE = 0 TRISD = 0 TRISC.0 = 0 TRISC. TRISC.3 = 0 adcon1 = 86 dato = 0 dato = i select case dato 'salidas 'Encender luces case 97 portd.0 = 1 case 98 portd.1 = 1 case 99 portd.2 = 1 case 100 portd.3 = 1 case 101 portd.4 = 1 </pre> | <pre> case 102 portd.5 = 1 case 103 case 99 portd.2 = 1 case 100 portd.3 = 1 case 101 portd.4 = 1 case 102 portd.5 = 1 case 103 portd.6 = 1 case 104 portd.7 = 1 'apagar luces case 105 portd.0 = 0 case 106 portd.1 = 0 case 107 portd.2 = 0 case 108 portd.3 = 0 case 109 portd.4 = 0 case 110 portd.5 = 0 </pre> |
|--|---|

Continuación de tabla V.

| | |
|---|---|
| <pre> case 111 portd.6 = 0 case 112 portd.7 = 0 'motores case 113 Portb.0 = 1 'porton abre Portb.1 = 0 case 114 Portb.0 = 0 'porton cierra Portb.1 = 1 i = 0 Portb = 0 portd = 0 Portc = 0 key = 1 main: USART_init(9600) ' se inicializa la comunicacion a 9600 Baudios while TRUE 'inicia el proceso i = USART_Read() ' recibe el dato case 115 'persiana de ventana sube Portb.2 = 1 Portb.3 = 0 case 116 'perciana de ventana baja </pre> | <pre> Portb.2 = 0 Portb.3 = 1 case 117 'persiana de ventana 2 sube Portb.4 = 1 Portb.5 = 0 case 118 'perciana de ventana 2 baja Portb.4 = 0 Portb.5 = 1 case 119 'puerta 1 Portb.6 = 1 case 120 Portb.6 = 0 case 121 'puerta 2 Portb.7 = 1 case 122 Portb.7 = 0 case 123 Portc.0 = 1 'regado de jardin case 124 Portc.0 = 0 end select if i = 67 then if key = 1 then numero = porta usart_write (numero) key = 0 end if end if if i = 48 then key = 1 end if Portb = i wend end. </pre> |
|---|---|

Fuente: elaboración propia.

4.3. Diseño de la aplicación Android para la interface hombre-máquina (HMI)

Para la interface entre el usuario y el sistema se utilizará un celular con sistema Android. Para esto es necesario realizar una aplicación que se pueda abrir en el teléfono celular y desde allí controlar y supervisar el sistema. Existen varios softwares para realizar las aplicaciones Android, algunas de ellas requieren un conocimiento de programación muy grande. Una opción para estudiantes, diseñadores de aplicaciones y personas que quieran realizar el diseño de una aplicación, es APP Inventor. Este software que utiliza el lenguaje de programación JAVA, es una plataforma en línea que hace más fácil la tarea de programación de aplicaciones Android.

4.3.1. Programación en lenguaje java a través del software APP Inventor

App Inventor es una herramienta basada en la nube, lo que significa que se pueden construir aplicaciones directamente en el navegador web. Este sitio web ofrece todo el apoyo que se necesita a medida que se aprende cómo construir aplicaciones. El software APP Inventor, es un servicio gratuito que se puede encontrar en la página www.ai2.appinventor.mit.edu.

4.3.2. Instalar y configurar el APP Inventor

Para trabajar con APP Inventor primero se instala y configura el paquete de herramientas de software, esto se hace ingresando a la página de APP Inventor y descargando el paquete; este software funciona en Windows XP, Vista y Windows 7. El APP inventor es un software que funciona en línea, solo funciona con el buscador Chrome y con Firefox.

Cuando ya se descargó el paquete, se procede a instalarlo en la computadora; para esto se hace clic a través de los pasos del instalador. No se debe cambiar la ubicación de la instalación, ya que es posible que más adelante se tengan que comprobar los controladores. El directorio será diferente dependiendo de la versión de Windows y si está o no está en el sistema como administrador. Es necesario tener instalado en la computadora la versión más actualizada del software JAVA; este programa se puede descargar desde la página oficial de java, <https://www.java.com/es/download/>.

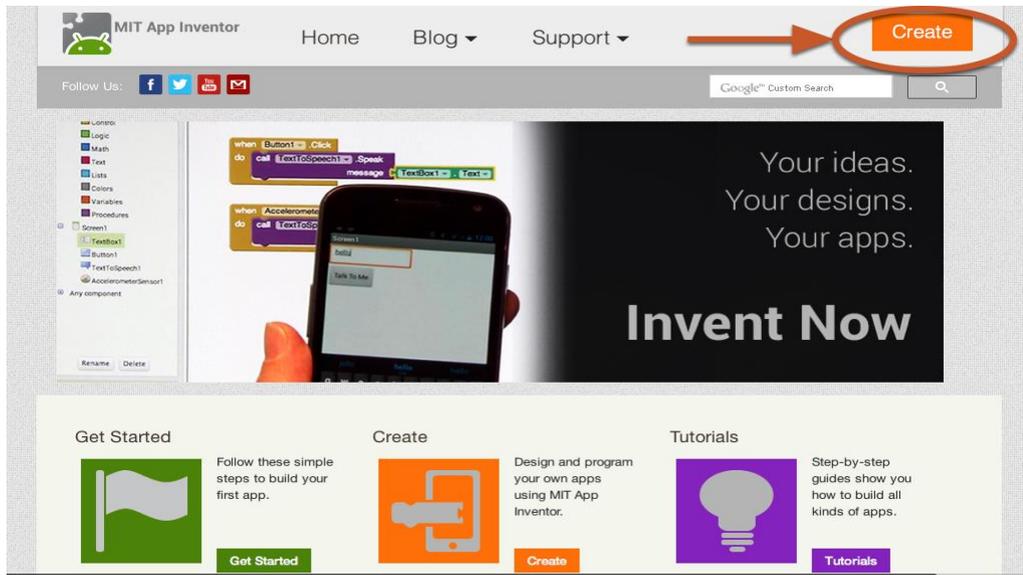
Se puede configurar la aplicación APP Inventor y empezar a construir aplicaciones en cuestión de minutos. El diseñador y editor de bloques se ejecutan completamente en el navegador. Para ver la aplicación en un dispositivo mientras se construye se realiza una prueba en vivo.

Se tienen tres opciones para comprobar el funcionamiento de la aplicación que se está diseñando. La primera opción es cuando se cuenta con un dispositivo Android con una conexión de internet vía wireless. La segunda opción es utilizar un simulador de dispositivo Android, que se puede descargar desde la página de APP Inventor. Y la tercera es conectar el dispositivo Android vía conexión USB.

4.3.3. Programar la aplicación con APP Inventor

Para iniciar con la programación de la aplicación se ingresa a la página <http://appinventor.mit.edu/explore/>, y se hace clic en crear; hay que ingresar un correo de gmail existente o si no se tiene, hay que hacer una cuenta de correo nuevo, para crear una sesión en APP Inventor.

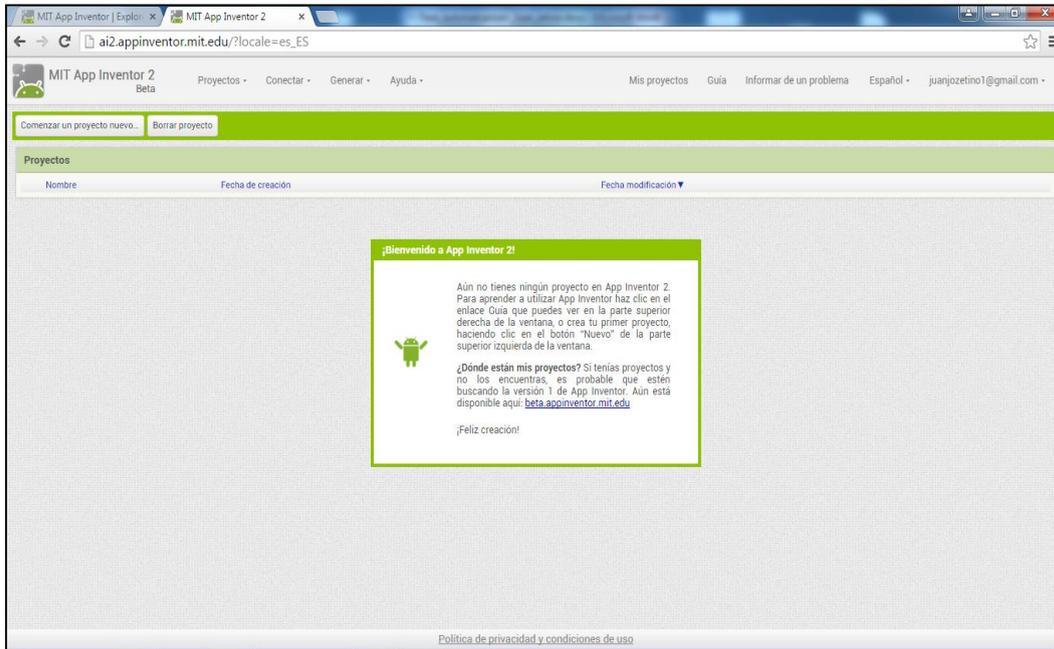
Figura 52. **Página de inicio de APP Inventor**



Fuente: MIT APP Inventor. *TalkToMe:Your first App Inventor app*. p. 1.

Ingresando el correo se abre la página principal que se muestra en la figura 53; en esa ventana se encuentran los proyectos que se han realizado: configurar la conexión con el dispositivo Android y generar los archivos para instalar la aplicación en el dispositivo Android; se puede escoger el idioma que se quiere utilizar. Para iniciar en la programación dar clic en comenzar un proyecto nuevo. Se muestra la ventana donde se coloca el nombre del proyecto. Este nombre no debe llevar espacios, sí se permite el uso de guión bajo.

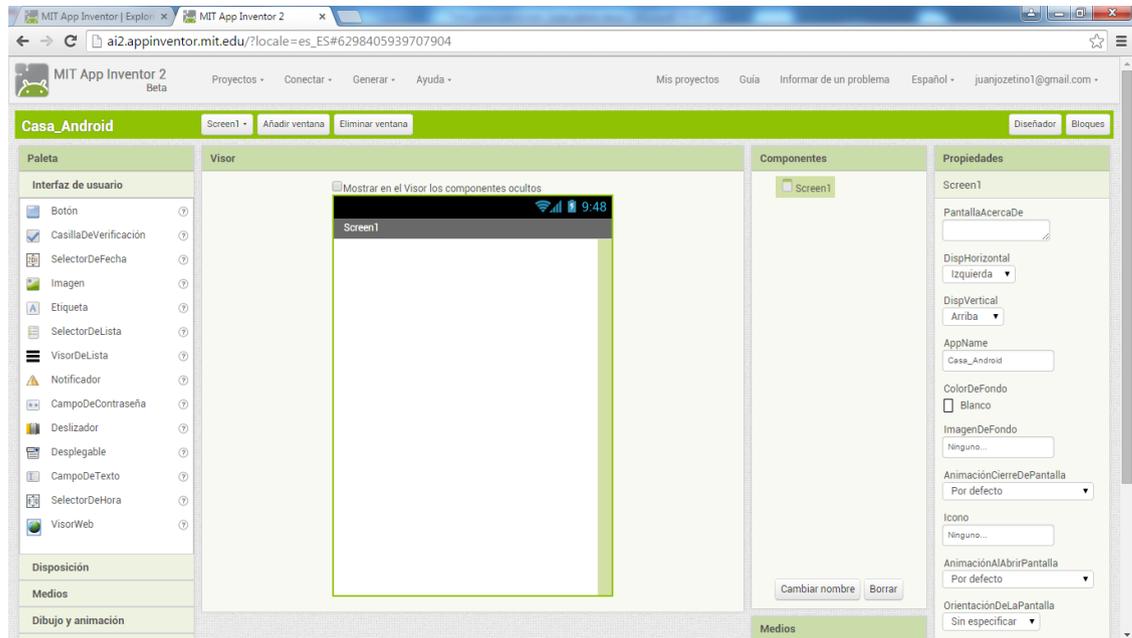
Figura 53. **Página de proyecto de APP Inventor**



Fuente: MIT APP Inventor. *TalkToMe: Your first App Inventor app*. p. 4.

Se abre la ventana de la figura 54. La ventana de diseño, o simplemente *designer* es donde se presenta el aspecto de la aplicación, y especificar qué funcionalidades debe tener. Se deben elegir los componentes que va tener el interfaz de usuario de la aplicación, como botones, imágenes, cuadros de texto, selector de listas, etiquetas, funcionalidades como *Text-to-Speech*, sensores, GPS o conexión Bluetooth.

Figura 54. **Ventana de diseño de usuario**

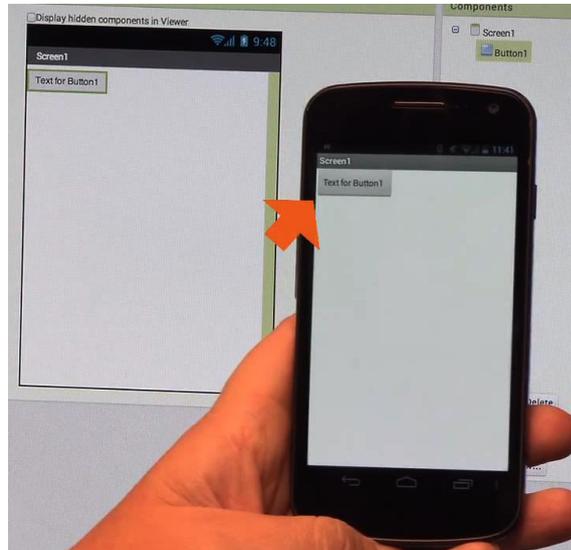


Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

Para iniciar con el diseño de una aplicación se organiza la pantalla; esto se logra con la herramienta disposición, y se puede disponer horizontal, vertical y de forma tabular. Esta herramienta sirve para organizar los elementos de la pantalla en forma de cuadrícula y que en cada cuadro se pueda ubicar un elemento. Así es más fácil organizar los elementos visibles de la aplicación, para darle un entorno o diseño ordenado a la aplicación.

De la paleta interfaz de usuario escoger un botón, se le da clic y se arrastra hacia la pantalla 1, en la posición elegida. Para visualizar la misma imagen en un dispositivo, hay que conectarlo a la computadora y hacer clic en el menú conectar. Escoger USB, para realizar la conexión vía puerto USB. Al realizar una conexión exitosa se muestra en el dispositivo la misma imagen que en la ventana diseñador, como se muestra en la figura 55.

Figura 55. Selección de botón y prueba en teléfono



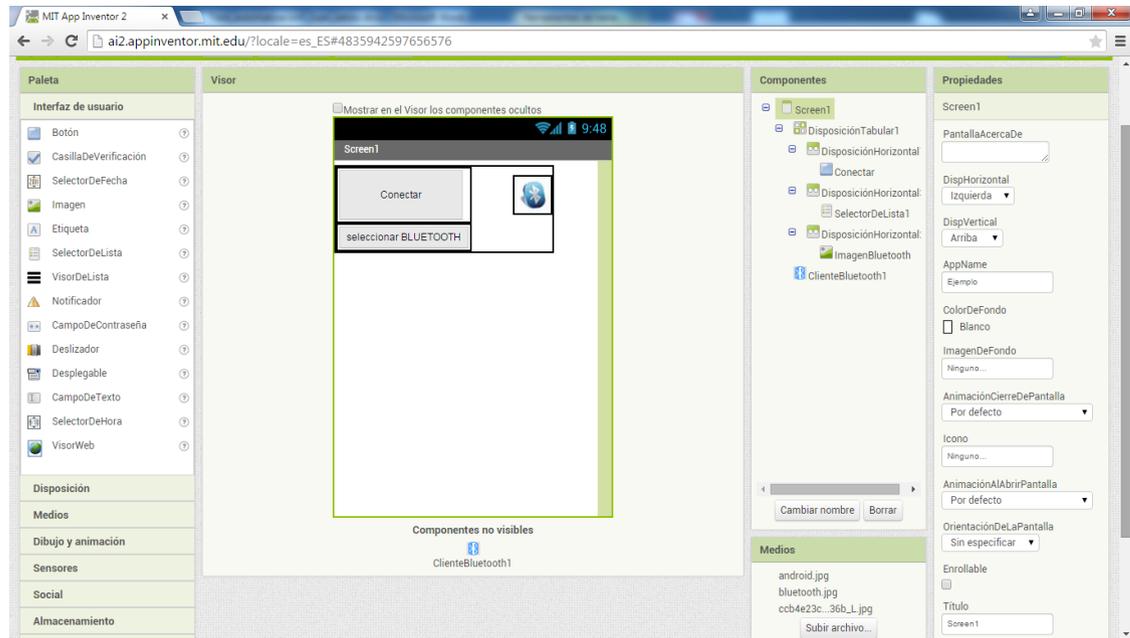
Fuente: MIT APP Inventor. *TalkToMe: Your first App Inventor app*. p. 11.

Ya con el botón en la pantalla, se selecciona y se va a la parte derecha de la misma a la paleta propiedades. En esta paleta se puede modificar el color de fondo; si está habilitado el tipo estilo y tamaño de letra, escoger una imagen para que lleve el botón, la forma, color del texto, ancho y altura del botón, entre otros.

Cada objeto que sea agregado en la pantalla del visor, va a ser enlistado en la paleta componentes. En esta paleta se le puede cambiar el nombre a cada objeto o borrarlo de la aplicación. Cualquier documento que se utilice, tal como imágenes, audio o video, se encontrará en la paleta medios, donde también se pueden subir archivos nuevos.

Hay objetos que no se muestran en la aplicación, entonces se muestran en componentes no visibles, como por ejemplo, un clienteBluetooth1. Que representa una conexión Bluetooth.

Figura 56. Pantalla de diseñador



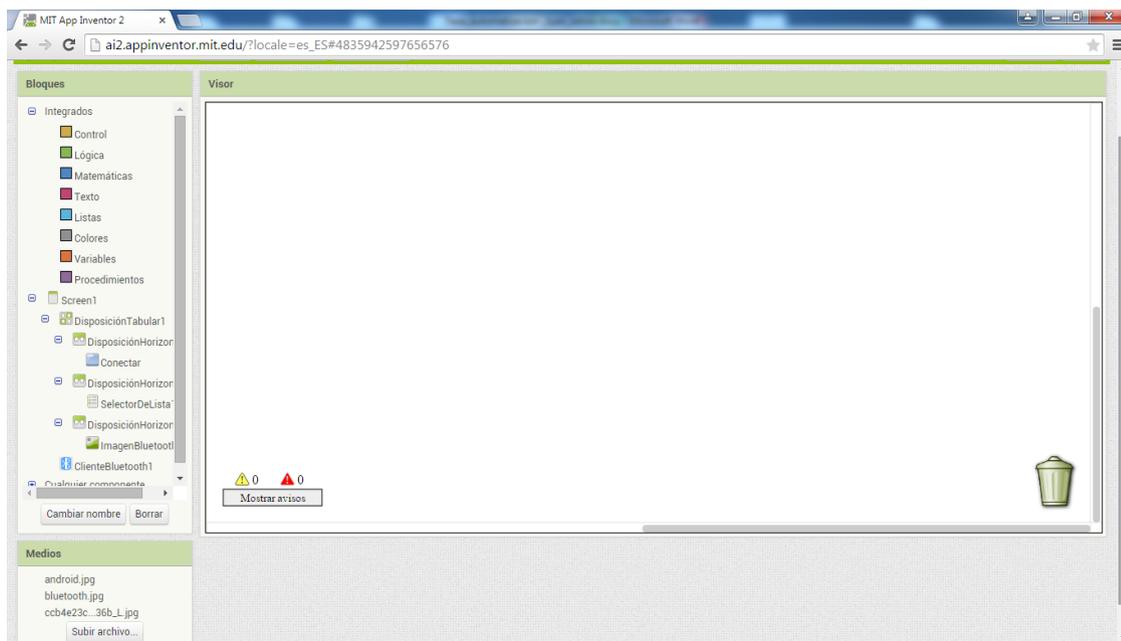
Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

Después de agregar los elementos con los que va contar la aplicación se le da el formato a cada uno de los elementos, utilizando para esto la paleta propiedades. Ya establecida la interface de la aplicación, se le da clic al botón bloques. Se abre una ventana distinta, que es el editor de bloques.

El editor de bloques es donde se programa el comportamiento de la aplicación. Hay bloques integrados que manejan operaciones como las matemáticas, la lógica, y el texto. Debajo de eso están los bloques que van con cada uno de los componentes de la aplicación. Con el fin de obtener los bloques para un determinado componente que aparezca en el editor de bloques, primero se tiene que añadir ese componente a su aplicación a través del diseñador. En la ventana de editor de bloques se encuentran dos partes importantes: las áreas de bloques y de visor. En los bloques se encuentran de

control, lógica, matemáticas, texto, listas, colores, variables y procedimientos. Cada elemento que se coloca en la ventana diseñador también se muestra en esta, pero acá se presentan en forma de bloques. En el visor se irán colocando los bloques del programa según se estructure, y se analizan los posibles errores cometidos, si los hay. En el visor también hay una papelera, que sirve para borrar bloques.

Figura 57. **Ventana de bloques de APP Inventor**

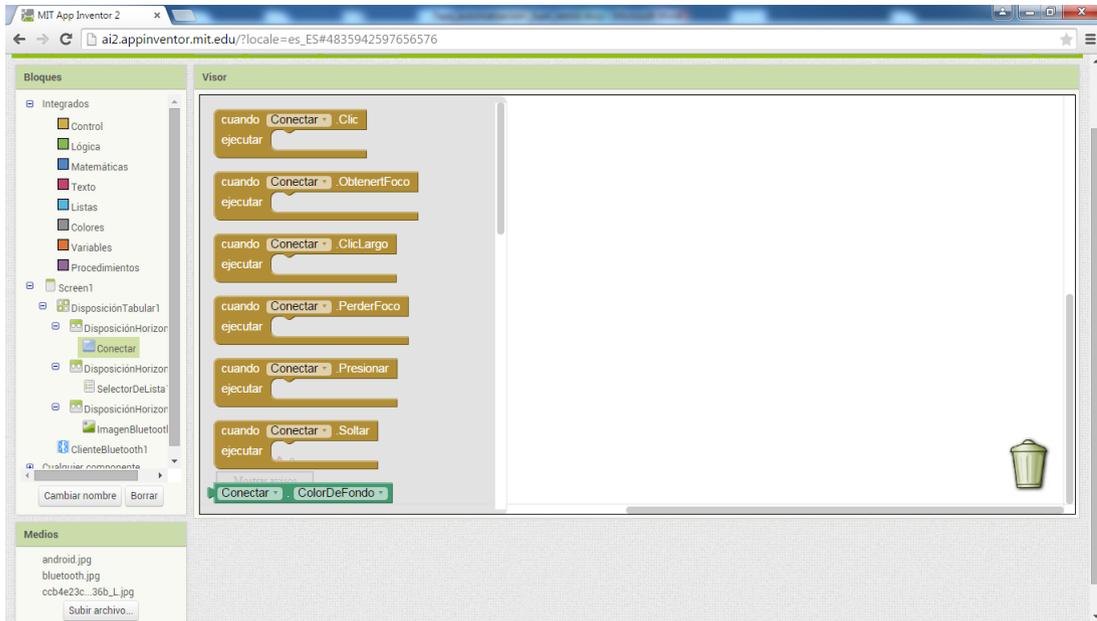


Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

Al hacer un clic en uno de los componentes se muestran todas las herramientas de programación que posee dicho componente, por ejemplo si se selecciona el botón conectar, se muestran las herramientas de programación, cuando hace clic, cuando suelta, cuando hace clic largo, entre otros; o modificar las propiedades como habilitar el botón conectar, poner visible el botón conectar o poner tamaño de letra en botón, entre otros. Estos bloques se muestran en

forma de rompecabezas, como aparecen en la figura 58. Esto hace que sea una forma muy sencilla de integrar el proyecto.

Figura 58. **Seleccionando bloques**



Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

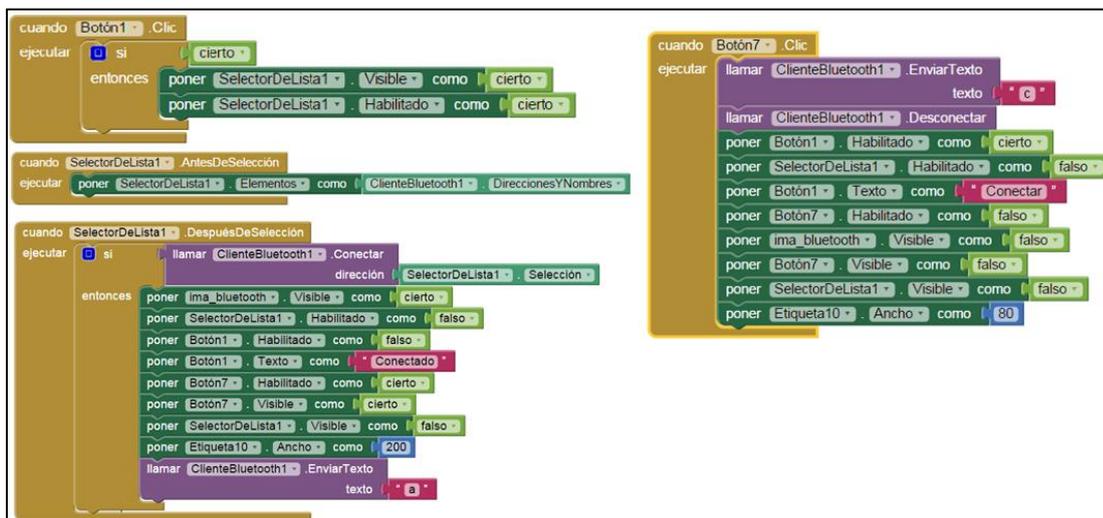
El diseño de la programación se inicia seleccionando los bloques con las funciones necesarias para que la aplicación realice la tarea deseada. En la figura 59 se muestra el diseño de la programación, para realizar la comunicación Bluetooth con el microcontrolador que realizará el control de actuadores y sensores. Se selecciona el bloque; cuando se hace clic en el botón 1, que se utiliza para conectar. El botón 1 hace que se habilite y sea visible una lista para seleccionar el dispositivo Bluetooth con el que se va a conectar el dispositivo Android. El bloque antes de la selección se utiliza para que, cuando el usuario le dé clic a la lista, esta muestre una posible lista de dispositivos Bluetooth con los que se puede conectar el dispositivo Android.

El bloque, después de la selección, se utiliza para activar la función conectar con cliente Bluetooth, indicándole la dirección que se seleccionó en la lista.

Si se realiza la comunicación, entonces se coloca de forma visible una imagen del símbolo de Bluetooth; se habilita el botón que se utiliza para desconectar la comunicación; se habilitan también los botones para seleccionar la función a controlar en la casa y también se envía un *byte* en forma de texto para que el microcontrolador muestre una señal, que indique que ya está conectado con el dispositivo Android.

El bloque, cuando el botón 7 hace clic, se utiliza para, enviar un byte en forma de texto al microcontrolador, para que se apague la señal de conectado. Activa la función desconectar, coloca visible el botón conectar, oculta la imagen del símbolo Bluetooth y oculta los botones para realizar las tareas de control.

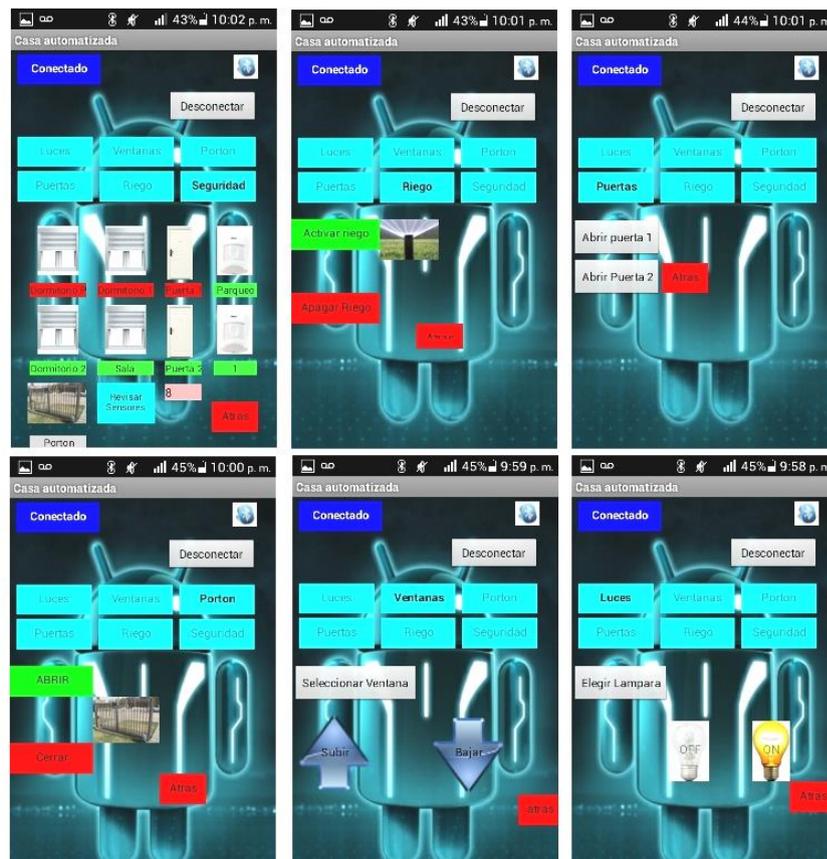
Figura 59. **Diseño de la comunicación Bluetooth**



Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

El diseño de toda la aplicación se realiza de esta forma; en la figura 60 se muestra el diseño de las pantallas; a través de la programación se hace cambiar la pantalla; hay bloques que colocan de forma visible y habilitan las distintas funciones de la aplicación. Al seleccionar una función se deshabilitan las otras. Cada una de las funciones va tener asociados bloques de programación, que se encargan de enviar información al microcontrolador, también se colocan de forma visible las pantallas, se habilitan los botones y se realizan operaciones matemáticas y lógicas, operaciones de comparación y control.

Figura 60. **Diseño de las distintas pantallas de la aplicación**

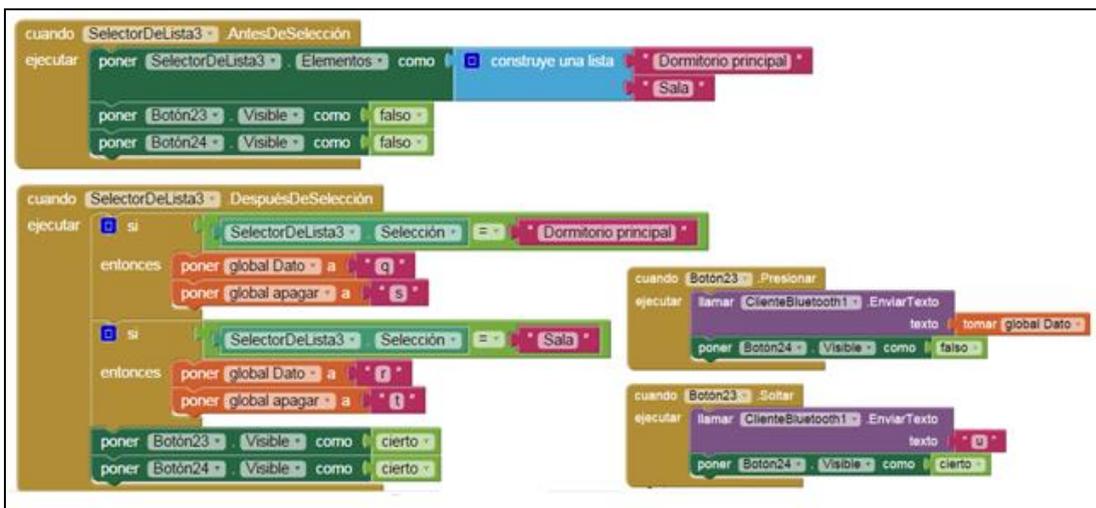


Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

Para hacer llegar la información hacia el microcontrolador, el programa le envía *bytes* en forma de texto, como por ejemplo las letras a y b. Estas letras viajan de bit en bit y llegan al microcontrolador, el cual cuando recibe la información realiza la tarea indicada en su programa principal. La figura 61 muestra los bloques que realizan el control de persianas de las dos ventanas de la casa.

En esta parte del programa se tiene una lista de selección, en la cual se crea una lista con los nombres de las dos ventanas de la casa; al seleccionar la ventana, se le carga un *byte* en forma de texto a dos variables globales, que van a servir para enviárselos al microcontrolador al presionar el botón 23, que se utiliza para subir la persiana. Para asegurar que no se puedan activar las dos funciones al mismo tiempo, bajar o subir la persiana, al presionar el botón 23 se deshabilita el botón 24, que se utiliza para la función bajar persiana.

Figura 61. Bloques del control de persianas para ventanas



Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

El control de luces de las puertas, de portón y el regado de jardín, se hacen de forma similar, enviando bytes al microcontrolador, el cual realiza una tarea que tiene programada.

En el control de seguridad es la única parte donde el microcontrolador envía datos a la aplicación. La figura 62 muestra los bloques de programación para el control de seguridad perimetral. Con el botón 26 se le envía un *byte* al microcontrolador, para indicarle que se quiere saber el estado de los sensores que están conectados a su periferia. El microcontrolador recibe ese dato y envía a la aplicación un byte, con la información de los sensores.

El programa de la aplicación recibe ese dato y lo guarda en una variable. Este *byte* se recibe de forma numérica siendo este valor de 0 a 255 en forma decimal. Para poder mostrar qué sensor está activado, se debe convertir el valor decimal a binario. Primero se toma la variable donde se guardó el dato enviado por el microcontrolador y se le aplica la operación, módulo de la división de la variable entre dos, para obtener un uno o cero.

También se divide la variable entre dos, y se aproxima al valor menor del resultado. Esto se repite ocho veces, para obtener los bits del *byte* enviado por el microcontrolador. Luego, si el módulo da como resultado uno, se le cambia el color de fondo a una etiqueta a color rojo, indicando que un sensor está activado. Por el contrario, si el resultado es cero, se cambia el color de fondo de la etiqueta del sensor a verde.

Figura 62. Bloques de programación para el control de seguridad

```
when Botón26 . TouchDown
do
  call ClienteBluetooth1 . SendText
  text " C "
  set global Numero to 0
  if call ClienteBluetooth1 . BytesAvailableToReceive > 0
  then
    set global Numero to call ClienteBluetooth1 . ReceiveUnsigned1ByteNumber
    set Etiqueta11 . Text to get global Numero
    set global resultado2 to 0
    set global var to get global Numero

when Botón26 . TouchUp
do
  call ClienteBluetooth1 . SendText
  text " 0 "
  while test get global resultado2 ≤ 7
  do
    set global resultado to modulo of get global var + 2
    set global var to floor get global var / 2
    if get global resultado = 0
    then
      if get global resultado = 1
      then
        set Etiqueta4 . BackgroundColor to red
      else
        set Etiqueta4 . BackgroundColor to green
    end if
  end if
  set global resultado2 to get global resultado + 1
  set Etiqueta11 . Text to get global resultado2
  set global Numero to 0
```

Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

4.3.4. Implementación de aplicación desarrollada en el teléfono celular

Al finalizar el diseño de la interface y programación de la aplicación, se descargan los archivos necesarios para instalar la aplicación en el teléfono. En el menú generar seleccionar la función, guardar archivo apk en el ordenador. Se inicia la descarga de los archivos desde la página de APP Inventor.

Al finalizar la descargase debe conectar el teléfono a la computadora y tener un programa para instalar aplicaciones. Se le da doble clic al archivo apk y se inicia la instalación de la aplicación. Al finalizar la instalación, la aplicación ya se puede utilizar.

4.4. Evaluación del funcionamiento del sistema de automatización

Para evaluar el funcionamiento del sistema se deben tener los sensores y actuadores conectados al circuito de potencia y al control principal, que a su vez ya se encuentre alimentado de voltaje. Entonces, la aplicación ya se encuentra instalada en el teléfono celular.

Como primer paso, hay que habilitar el Bluetooth del teléfono y abrir la aplicación. Una vez abierta la aplicación hacer clic en el botón conectar, se despliega una lista con los dispositivos Bluetooth disponibles para conectar; se selecciona el indicado. Para este caso, el módulo de comunicación tiene un nombre, que se dio en la configuración del módulo. Al seleccionar el módulo de comunicación, la aplicación inicia el proceso de conexión.

Cuando se establece la comunicación, la pantalla muestra una imagen del símbolo de Bluetooth y se habilita el botón desconectar. También se habilita el

menú principal con las distintas funciones del control. Al seleccionar cada menú se muestra una nueva pantalla.

En el control principal, se enciende una salida en RC2 que indica la conexión. El módulo de comunicación tiene un led que indica el estado; cuando está esperando conexión, parpadea a 1 HZ, y cuando se establece la comunicación se queda encendido.

Figura 63. **Conexión con el modulo Bluetooth**



Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

4.4.1. **Iluminación de la casa**

Al seleccionar el menú “luces”, cambia la pantalla de la aplicación. En esta pantalla se muestra una lista que indica la ubicación de las luces de la casa que se pueden encender o apagar. Cuando se elige una luz en la lista, se puede encender o apagar, mediante los botones que muestra la imagen de una bombilla apagada y encendida. Para cambiar de luz se debe ir a la lista de nuevo y seleccionar otra luz. Cuando se quiere regresar al menú anterior, se

puede hacer utilizando el botón “atrás”. Se pueden dejar encendidas las luces y desconectar la aplicación o cerrarla y no afecta el funcionamiento del control, debido que el microcontrolador mantiene su memoria y la guarda hasta el momento de conectarse de nuevo, y enviarle otra instrucción.

En el módulo de potencia se conecta la luz en las dos borneras que están indicadas para esta función. La luz puede ser de cualquier tipo y el voltaje que se va utilizar puede ser 120/240 VAC.

Figura 64. Iluminación de la casa



Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

4.4.2. Apertura de puertas

Las cerraduras de las puertas deben mantener energía durante todo el tiempo y al querer abrir la puerta se debe cortar la energía. La puerta también

debe de contar con un brazo hidráulico para que mantenga la puerta cerrada, y que la cerradura magnética se encargue de asegurar su cierre. Al seleccionar el menú puertas, cambia la pantalla y muestra dos botones que permiten abrir la puerta seleccionada.

El control le mantiene una señal constante al circuito de potencia para que este energice la cerradura en todo tiempo. Al momento de presionar el botón abrir, se le envía una señal al control para que se le quite la energía a la cerradura y se pueda abrir la puerta. Las cerraduras magnéticas tienen una bobina que trabajan con 24 VDC. Por lo que se debe utilizar el circuito de potencia con transistor.

Figura 65. **Control de cerraduras magnéticas**

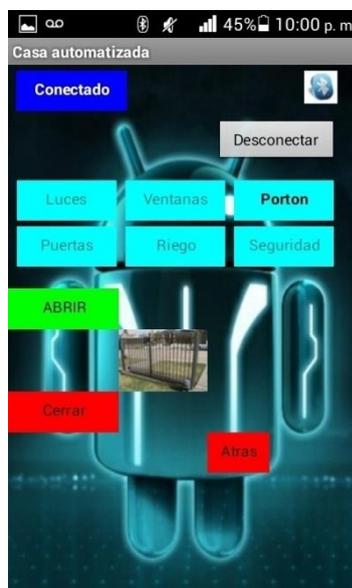


Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

4.4.3. Portón eléctrico

La apertura y cerrado del portón la realiza el motor eléctrico que está conectado al mecanismo. Selecciona el menú portón; en esta pantalla se muestran dos botones: uno para abrir y otro para cerrar el portón. Estos le envían datos al control para que este active el circuito de potencia del motor. Para abrir el portón, el motor se mantiene activo girando en un solo sentido, hasta que encuentra un interruptor de fin de carrera que está instalado en su sistema interno del mecanismo, que le indica que ya abrió el portón y esto hace que el motor se detenga; de igual forma para cerrar el portón, con la diferencia que se activa el giro contrario del motor. En la aplicación solo es posible activar una de las dos acciones al mismo tiempo. El motor del portón utiliza 120/240 VAC para su alimentación y por lo mismo el circuito de potencia utiliza un triac para el manejo de corriente alterna.

Figura 66. Apertura y cerrado del portón



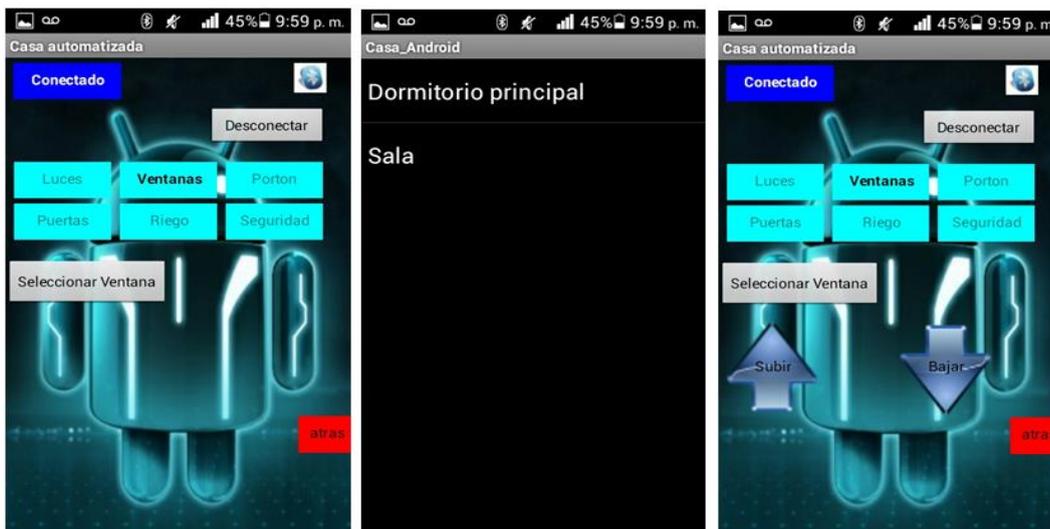
Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

4.4.4. Apertura de persianas de ventanas

Para bajar o subir las persianas se utiliza un motor de corriente alterna que está conectado al sistema mecánico de la persiana o cortina. Al seleccionar el menú “ventana”, se muestra una lista para escoger en cuál de las ventanas, se quiere controlar la persiana. Al seleccionar una de las dos ventanas aparecen dos botones que muestran la imagen de una flecha para subir o bajar la persiana. Estos le envían información al control para que activen el circuito de potencia del motor y este suba o baje la persiana a criterio de la persona.

El motor cuenta con un interruptor fin de carrera que le indica al motor que ya está arriba o abajo la persiana. Esto le quita la energía al motor para que no siga girando, aunque el usuario le siga enviando una señal. Los motores de las persianas utilizan 120/240 VAC. Por lo que se va a utilizar el circuito de potencia para motor alterno.

Figura 67. Control de motor de persiana de ventana



Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

4.4.5. Regado de jardín

Instalar el sistema hidráulico para el regado de jardín dependerá solo de la apertura de la electroválvula que permitirá el paso de agua hacia los aspersores que tienen como trabajo el regado de las áreas del jardín. Al seleccionar la función riego se muestran en la pantalla dos botones: apagar y encender riego, los cuales envían señal al control para que active o desactive la electroválvula que está conectada al circuito de potencia. La electroválvula tiene un solenoide de 24VDC. El usuario decide cuándo activar el riego y cuándo apagarlo, o puede programar el tiempo que quiere que esté activado el riego.

Figura 68. **Activación de electroválvula de riego**



Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

4.4.6. Seguridad perimetral

La casa cuenta con sensores de movimiento y sensores magnéticos en las puertas y ventanas, para indicar el estado de la puerta o ventana. Seleccionando la función seguridad, se muestran las distintas partes de la casa que pueden ser controladas. Se habilita un botón que va a indicar el estado de los sensores de la casa. En la pantalla aparece la imagen de puertas, ventanas y sensores. Cada uno de los sensores tiene su nombre en la parte inferior.

Al momento de activar el botón que revisa los sensores, el control le envía una respuesta al teléfono indicándole el estado de los sensores. Si por ejemplo el sensor de la puerta principal está activo, indicando que la puerta está abierta, en la imagen aparece el nombre con un fondo de color rojo y si está desactivo el fondo se muestra de color verde. Se puede programar que si está activa una alarma, esta genere una señal que haga que vibre el teléfono o genere un sonido, para indicar que algún sensor está activo.

Los sensores de movimiento tienen que ser graduados para que no detecten mascotas, y el tiempo que permanece activa la señal de movimiento. Los sensores se conectan directos al control porque todos tienen salida a contacto N.O o N.C. por lo que no necesita ningún tipo de convertidor a 5VDC.

Figura 69. **Control de seguridad perimetral de la casa**



Fuente: elaboración propia, empleando la plataforma en línea APP Inventor.

4.5. Costos del sistema propuesto

Los costos de la instalación de un sistema domótico en general son elevados, debido a que los precios de los componentes, los controles principales, interfaces humano-máquina son muy altos. En muchos casos no se cuenta con estos productos en nuestro país, por lo que hay necesidad de comprarlos en el exterior, y si a esto se le agrega que las casas no han sido diseñadas para instalarles estos sistemas. Por lo que hay que modificar la estructura física o agregar nuevos ductos o instalar nuevo cableado en toda la casa. Pero al ver lo fácil que hace la vida cotidiana el tener instalado un sistema domótico, se compensan los gastos.

4.5.1. Costos de los materiales

Los materiales para la instalación de este sistema se pueden dividir entre los componentes necesarios para el control, los sensores, actuadores, y para el

cableado. Pero por tener como interface entre la máquina y el humano, un teléfono celular hace reducir costos, debido que el costo de una pantalla táctil o algún tipo de control gráfico es elevado.

Tabla VI. **Costos de materiales para el control**

| Correlativo | Descripción del material | Cantidad | Costo en quetzales | Costo total |
|-------------|----------------------------------|----------|--------------------|-----------------|
| 1 | Microcontrolador 16F877A | 1 | 50,00 | 50,00 |
| 2 | Módulo de comunicación Bluetooth | 1 | 450,00 | 450,00 |
| 3 | Reguladores de voltaje | 2 | 5,00 | 10,00 |
| 4 | Resistencias de 330 ohms ¼ W | 100 | 0,25 | 25,00 |
| 5 | Led de 3 mm color verde | 100 | 1,00 | 100,00 |
| 6 | Borneras de dos tornillos | 50 | 3,00 | 150,00 |
| 7 | Tarjeta para circuito impreso | 1 | 100,00 | 100,00 |
| 8 | Optotriac | 15 | 5,00 | 75,00 |
| 9 | Triac de 8 A. | 15 | 10,00 | 150,00 |
| 10 | Transistores TIP41 | 5 | 5,00 | 25,00 |
| 11 | Fuente de alimentación protegida | 1 | 300,00 | 300,00 |
| 12 | Caja de metal para proyecto | 1 | 100,00 | 100,00 |
| | Total | | | 1 685,00 |

Fuente: elaboración propia.

Tabla VII. **Costos de los sensores y actuadores**

| Correlativo | Descripción del material | Cantidad | Costo en quetzales | Costo total |
|-------------|-------------------------------|----------|--------------------|-----------------|
| 1 | Sensor magnético | 8 | 20,00 | 160,00 |
| 2 | Sensor de movimiento | 2 | 200,00 | 400,00 |
| 3 | Motor de portón | 1 | 2500,00 | 2 500,00 |
| 4 | Cerraduras magnéticas | 2 | 500,00 | 1 000,00 |
| 5 | Sistema de riego eléctrico | 1 | 500,00 | 500,00 |
| 6 | Sistema de persiana eléctrica | 2 | 1000,00 | 2 000,00 |
| | Total | | | 6 500,00 |

Fuente: elaboración propia.

Tabla VIII. **Costo de materiales para el cableado**

| Correlativo | Descripción del material | Cantidad | Costo en quetzales | Costo total |
|-------------|-------------------------------|----------|--------------------|-----------------|
| 1 | Cable calibre 18 color rojo | 2 rollo | 105,00 | 210,00 |
| 2 | Cable calibre 18 color negro | 2 rollo | 105,00 | 210,00 |
| 3 | Cable calibre 18 color blanco | 2 rollo | 105,00 | 210,00 |
| 4 | Cable calibre 18 color azul | 2 rollo | 105,00 | 210,00 |
| 5 | Cable calibre 14 color blanco | 2 rollo | 200,00 | 400,00 |
| 6 | Cable calibre 14 color negro | 2 rollo | 200,00 | 400,00 |
| 7 | Cintas de aislar | 5 rollos | 30,00 | 150,00 |
| | | | | 1 790,00 |

Fuente: elaboración propia.

4.5.2. **Costos de la fabricación del control e instalación**

La fabricación del control y la futura instalación del sistema también tendrían un costo para el proyecto. Pensando que se debe instalar todo el cableado, los sensores, actuadores y el control. Se debe en pensar cuando costaría ese trabajo.

Tabla IX. **Costo de fabricación e instalación del sistema**

| Correlativo | Descripción del trabajo | Cantidad | Costo en quetzales | Costo total |
|-------------|--------------------------------------|----------|--------------------|-----------------|
| 1 | Fabricación del control | 1 | 2 000,00 | 2 000,00 |
| 2 | Cableado de sensores y actuadores | 1 | 2 000,00 | 2 000,00 |
| 3 | Instalación de sensores y actuadores | 1 | 3 000,00 | 3 000,00 |
| 4 | Instalación del control | 1 | 1 000,00 | 1 000,00 |
| 5 | Diseño de la aplicación | 1 | 1 000,00 | 1 000,00 |
| | Total | | | 9 000,00 |

Fuente: elaboración propia.

El costo en la instalación y el cableado tendría que depender en mucho de las distancias de los componentes a instalar y sí se tiene que modificar la estructura de la casa.

4.5.3. Costo total del proyecto

Ya teniendo el desglose de todos los gastos que se tendrían que hacer para instalar un sistema de este tipo, se puede hacer un cálculo del precio total de un sistema de este tipo.

Tabla X. Costo total del proyecto

| Correlativo | Descripción del trabajo | Costo total |
|-------------|--|------------------|
| 1 | Costos de materiales para el control | 1 685,00 |
| 2 | Costos de los sensores y actuadores | 6 500,00 |
| 3 | Costo de materiales para el cableado | 1 790,00 |
| 4 | Costo de fabricación e instalación del sistema | 9 000,00 |
| | Total | 18 975,00 |

Fuente: elaboración propia.

El costo aproximado del proyecto sería de Q18 975,00; es un precio elevado, pero si se compara con otros sistemas puede resultar que sea un precio bajo. Este valor también depende del tipo de casa a instalar y de los precios de los componentes que se tengan que comprar en el exterior.

CONCLUSIONES

1. Los sistemas de automatización de casas hacen que sus funciones sean más fáciles de controlar y le proporciona más seguridad al habitante de la misma. Así también proporciona ahorro energético.
2. El control principal del sistema, debe contar con un circuito de mando, cuyo principal componente va ser el microcontrolador, el cual va tener conectados los sensores y circuitos de potencia que van a controlar las corrientes de los actuadores.
3. Para que pueda existir un enlace entre el control principal y un celular se utiliza el módulo de comunicación Bluetooth; este realiza una conversión entre comunicación Bluetooth y comunicación serial.
4. El diseño de una aplicación Android, utilizando APP Inventor, se inicia agregando y organizando todos los elementos de la interface de usuario, para que después estos se interconecten utilizando los bloques de función, para que la aplicación realice una tarea deseada.
5. La forma de programación en APP Inventor es gráfica, lo cual hace que con bloques se realice la tarea de programación; la programación también no es secuencial, sino que es paralela. Lo que hace que todos los bloques se ejecuten al mismo tiempo.

6. La aplicación le envía bytes de información al control principal, mediante el módulo de comunicación que le transmite los datos al microcontrolador; al recibir la información este procesa la información utilizando el programa que reside en su memoria y realiza la tarea de control en la casa que tiene programada.

7. La aplicación Android se puede hacer funcionar en teléfonos celulares y tablets que cuenten con Bluetooth integrado y una pantalla lo suficientemente grande como para visualizar los botones y las interfaces de usuario en la aplicación.

8. La distancia de comunicación entre el teléfono celular y el control. depende de la potencia de recepción del módulo Bluetooth, y de la potencia de que dispone el teléfono celular. Las paredes de la casa afectan en la comunicación, pero al disponer de un módulo de comunicación potente, se pueden lograr comunicaciones hasta 100 metros de distancia.

RECOMENDACIONES

1. Para que un sistema domótico funcione de mejor forma, es preferible que en el proceso de la construcción de la casa se contemplen los ductos para el cableado de los sensores, actuadores y control.
2. Los técnicos encargados deben tomar en cuenta que es preferible instalar sensores y actuadores que funcionen con voltajes iguales; los voltajes pueden ser 120VAC y 24VDC, debido que varios fabricantes utilizan estos valores.
3. En la elección de elementos para el control, hay que pensar en el ahorro energético, por lo que es preferible buscar los componentes que no consuman demasiada potencia eléctrica.
4. Se debe contar con un sistema de respaldo de energía como UPS o banco de baterías, para evitar que el sistema se bloquee o se desconecte por el corte de suministro eléctrico.
5. Es importante que las lámparas, puertas y portón eléctrico, cuenten con un interruptor o pulsador que haga que el apagado o encendido sea de forma manual, en el caso de algún desperfecto en el sistema.
6. Es importante que el módulo de comunicación Bluetooth cuente con al menos 100 metros de alcance, para que no se pierda la comunicación con el teléfono celular.

7. Para el funcionamiento adecuado de sensores, actuadores y circuitos de control, es importante el mantenimiento preventivo, para evitar que con el uso se desconecte algo y correctivo en caso de fallas.
8. Para el manejo de APP Inventor es importante conocerlos conceptos básicos de programación, asimismo contar con una conexión a internet por tratarse de una plataforma en línea.
9. Para el funcionamiento de la aplicación se puede utilizar un teléfono celular con sistema Android desde una versión 1,5, pero lo ideal sería contar con una versión 2,2 en adelante.
10. Es importante instalar la aplicación en al menos dos dispositivos distintos, evitando así perder el control en caso de una falla. Asimismo, se recomienda guardar los archivos APK en una computadora.
11. El actuador a utilizar para controlar una parte de la casa va a depender del espacio que se tiene para su instalación, que le dé seguridad al sistema, que represente una ayuda al usuario y que genere un ahorro energético para la casa.

BIBLIOGRAFÍA

1. ABELSON, Hal; LOONEY, Liz; SPERTUS, Ellen; WOLBER, David. *App Inventor 2*. 2a ed. U.S.A.: O´reilly media, 2014. 327 p. ISBN: 978-1491906842.
2. BALCELLS, Sendra; ROMERAL, Josep; JOSÚ, Luí. *Autómatas programables*. España: MARCOMBO, 2000. 450 p. ISBN: 8426710891.
3. BOYLESTAD, Robert. L. *Introducción al análisis de circuitos*. 10a ed. México: Pearson Educación, 2004. 1248 p. ISBN: 9702604486.
4. ORTIZ YÁÑEZ, Rubén. *Control eléctrico en los sistemas de edificios inteligentes*. México: Instituto Politécnico Nacional Tres Guerras, 2006. 323 p.
5. STEVENS, Roger. *PIC microcontroller serial communications from Square 1*. U.S.A.: Kelseyville, 1999. 526 p. ISBN: 0965416224 9780965416221.
6. TOMÁS GIRONÉS, Jesús. *El gran libro de Android*. 2a ed. España: MARCOMBO, 2012. 404 p. ISBN: 8426718329.
7. VERLE, Milan. *PIC Microcontrollers - Programming in Basic*. Zemun, Belgrade: mikroElektronika, 2010. 346 p. ISBN-13: 978-86-84417-18-5.

ANEXOS

Anexo 1. Organización de la memoria RAM del PIC16F877A

| | | | | | | | |
|---|-----|---|--------------|---|---|---|---|
| INDF | 00h | INDF | 80h | INDF | 100h | INDF | 180h |
| TMR0 | 01h | OPTION_REG | 81h | TMR0 | 101h | OPTION_REG | 181h |
| PCL | 02h | PCL | 82h | PCL | 102h | PCL | 182h |
| STATUS | 03h | STATUS | 83h | STATUS | 103h | STATUS | 183h |
| FSR | 04h | FSR TRISA | 84h | FSR | 104h | FSR | 184h |
| PORTA | 05h | TRISA | 85h | | 105h | | 185h |
| PORTB | 06h | TRISB | 86h | PORTB | 106h | TRISB | 186h |
| PORTC | 07h | TRISC | 87h | | 107h | | 187h |
| PORTD | 08h | TRISD | 88h | | 108h | | 188h |
| PORTE | 09h | TRISE | 89h | | 109h | | 189h |
| PCLATH | 0Ah | PCLATH | 8Ah | PCLATH | 10Ah | PCLATH | 18Ah |
| INTCON | 0Bh | INTCON | 8Bh | INTCON | 10Bh | INTCON | 18Bh |
| PIR1 | 0Ch | PIE1 | 8Ch | EEDATA | 10Ch | EECON1 | 18Ch |
| PIR2 | 0Dh | PIE2 | 8Dh | EEADR | 10Dh | EECON2 | 18Dh |
| TMR1L | 0Eh | PCON | 8Eh | EEDATH | 10Eh | Reservado | 18Eh |
| TMR1H | 0Fh | | 8Fh | EEADRH | 10Fh | Reservado | 18Fh |
| T1CON | 10h | | 90h | | 110h | | 190h |
| TMR2 | 11h | SSPCON2 | 91h | Registros de Propósito General 16 Bytes | Registros de Propósito General 16 Bytes | Registros de Propósito General 16 Bytes | Registros de Propósito General 16 Bytes |
| T2CON | 12h | PR2 | 92h | | | | |
| SSPBUF | 13h | SSPADD | 93h | | | | |
| SSPCON | 14h | SSPSTAT | 94h | | | | |
| CCPR1L | 15h | | 95h | | | | |
| CCPR1H | 16h | | 96h | | | | |
| CCP1CON | 17h | | 97h | | | | |
| RCSTA | 18h | TXSTA | 98h | | | | |
| TXREG | 19h | SPBRG | 99h | | | | |
| RCREG | 1Ah | | 9Ah | | | | |
| CCPR2L | 1Bh | | 9Bh | | | | |
| CCPR2H | 1Ch | | 9Ch | | | | |
| CCP2CON | 1Dh | | 9Dh | | | | |
| ADRESH | 1Eh | ADRESL | 9Eh | | | | |
| ADCON0 | 1Fh | ADCON1 | 9Fh | | | | |
| | 20h | | A0h | | | | |
| Registros de Propósito General 96 Bytes | | Registros de Propósito General 80 Bytes | 0EFh 0F0h | 120h | 120h | 1A0h | 1A0h |
| | | | | 16Fh 170h | 16Fh 170h | 1EFh 1F0h | 1EFh 1F0h |
| Banco 0 | 7Fh | Banco 1 | FFh | Banco 2 | 17Fh | Banco 3 | 1FFh |

Fuente: elaboración propia, empleando Paint.

