



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

EXTENSIÓN LATACUNGA

CARRERA DE INGENIERÍA EN ELECTRÓNICA E
INSTRUMENTACIÓN

“DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE CONTROL, MONITOREO Y VIGILANCIA POR CÁMARA WEB, VIA INTERNET, PARA LA ALIMENTACIÓN DE ANIMALES DOMÉSTICOS”.

Christian Geovanny Masaquiza Criollo

Luis Orlando Saquinga Sailema

Tesis presentada como requisito previo a la obtención del grado de:

INGENIERO EN ELECTRÓNICA E INSTRUMENTACIÓN

Año 2013

**UNIVERSIDAD DE LA FUERZAS ARMADAS – ESPE
EXTENSIÓN LATACUNGA**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA E
INSTRUMENTACIÓN**

DECLARACIÓN DE RESPONSABILIDAD

Nosotros, Christian Geovanny Masaquiza Criollo y Luis Orlando Saquinga
Sailema

DECLARO QUE:

El proyecto de grado denominado “DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE CONTROL, MONITOREO Y VIGILANCIA POR CÁMARA WEB, VIA INTERNET, PARA LA ALIMENTACIÓN DE ANIMALES DOMÉSTICOS”, ha sido desarrollado en base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan el pie de las páginas correspondiente, cuyas fuentes se incorporan en la bibliografía. Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.

Latacunga, Noviembre del 2013.

Christian Geovanny Masaquiza Criollo

Luis Orlando Saquinga Sailema

CI: 1803754223

CI: 1804025094

**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
EXTENSIÓN LATACUNGA**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA E
INSTRUMENTACIÓN**

CERTIFICADO

Ing. José Bucheli A. (DIRECTOR)
Ing. David Rivas L. (CODIRECTOR)

CERTIFICAN

Que el trabajo titulado “DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE CONTROL, MONITOREO Y VIGILANCIA POR CÁMARA WEB, VIA INTERNET, PARA LA ALIMENTACIÓN DE ANIMALES DOMÉSTICOS”, realizado por Christian Geovanny Masaquiza Criollo y Luis Orlando Saquina Sailema ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Universidad de las Fuerzas Armadas-ESPE.

Debido a que constituye un trabajo de excelente contenido científico que coadyuvará a la aplicación de conocimientos y al desarrollo profesional SI recomiendan su publicación.

El mencionado trabajo consta de dos documentos empastados y dos discos compactos el cual contiene los archivos en formato portátil de Acrobat (pdf). Autorizan a Christian Geovanny Masaquiza Criollo y Luis Orlando Saquina Sailema que lo entregue a Ing. José Bucheli en su calidad de Director de la Carrera.

Latacunga, Noviembre del 2013.

Ing. José Bucheli A.
DIRECTOR

Ing. David Rivas L.
CODIRECTOR

**UNIVERSIDAD DE LAS FUERZAS ARMADAS - ESPE
EXTENSIÓN LATACUNGA**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA E
INSTRUMENTACIÓN**

AUTORIZACIÓN

Nosotros, Christian Geovanny Masaquiza Criollo y Luis Orlando Saquina Sailema.

Autorizamos a la Universidad de las Fuerzas Armadas la publicación, en la biblioteca virtual de la Institución del trabajo “DISEÑO Y CONSTRUCCIÓN DE UN SISTEMA DE CONTROL, MONITOREO Y VIGILANCIA POR CÁMARA WEB, VIA INTERNET, PARA LA ALIMENTACIÓN DE ANIMALES DOMÉSTICOS”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría.

Latacunga, Noviembre del 2013.

Christian Geovanny Masaquiza Criollo

CI: 1803754223

Luis Orlando Saquina Sailema

CI: 1804025094

DEDICATORIA

Christian Geovanny Masaquiza Criollo

A ti Dios que me diste la oportunidad de vivir y regalarme una familia maravillosa.

Con mucho cariño y amor principalmente a mis padres Ernesto y Angélica que me dieron la vida y han estado conmigo en todo momento para que yo pudiera lograr mis sueños y por creer en mí, aunque hemos pasado momentos difíciles siempre han estado apoyándome y brindándome todo su amor y afecto, por todo esto les agradezco el que estén conmigo a mi lado.

A tu paciencia y comprensión, preferiste sacrificar tu tiempo para que yo pudiera cumplir con lo mío. Por tu bondad y sacrificio me inspiraste a ser mejor para ti, ahora puedo decir que esta tesis lleva mucho de ti, gracias por estar siempre a mi lado, querida Esposa Sandra, mi dulce Hija Naomi y Mama Georgina.

A mis Hermanas Mercy y Alexandra, que han estado en las buenas y en las malas y que nunca me han dejado solo, gracias por estar conmigo siempre y apoyarme en todo las quiero mucho, mis sobrinos Michael, Nuria y mi cuñado Walter, gracias por creer en mí y apoyarme.

A Germán Lalaleo que ya no está entre nosotros, pero que está junto a DIOS, que fue más que un amigo, como un hermano para mí.

Dedico a esas personas importantes en mi vida, que siempre estuvieron listas para brindarme toda su ayuda, ahora me toca regresar un poquito de todo lo inmenso que me han otorgado. Con todo mi cariño esta tesis se las dedico a ustedes: Abuelitos, Tíos, Tías, Primos, Primas y Amigos.

DEDICATORIA

Luis Orlando Saquinga Sailema

Mi tesis la dedico con todo mi amor y cariño.

A ti mi DIOS quién supo siempre guiarme por el buen camino, darme fuerzas para seguir adelante y no desmayar en los problemas que se presentaban, enseñándome a encarar las adversidades sin perder nunca la dignidad ni desfallecer en el intento.

A mi madre Martha, por ser el pilar más importante y por demostrarme siempre su cariño y apoyo incondicional sin importar nuestras diferencias de opiniones. A mi padre Manuel por todos sus consejos, apoyo, comprensión, amor, ayuda en los momentos difíciles, y por ayudarme con los recursos necesarios para estudiar. A mi abuelita Teresa que ha sido como una segunda madre para mí que ha estado a mi lado en aquellos momentos difíciles de mi vida.

A mis familiares, viejos amigos y a quienes recién se sumaron a mi vida para hacerme compañía con sus sonrisas de ánimo, en especial a ti hermanita querida Estefanía, porque a lo largo de este trabajo aprendimos que nuestras diferencias se convierten en riqueza cuando existe respeto y verdadera amistad.

AGRADECIMIENTO

Christian Geovanny Masaquiza Criollo

Le agradezco a DIOS por haberme acompañado y guiado a lo largo de mi carrera, por ser mi fortaleza en los momentos de debilidad y por brindarme una vida llena de aprendizajes, experiencia y sobre todo felicidad y salud.

Le doy gracias a mis padres Ernesto y Angélica por apoyarme en todo momento, por los valores que me han inculcado, y por haber dado la oportunidad de tener una excelente educación en el trascurso de mi vida. Sobre todo por ser un excelente ejemplo de vida a seguir.

Le agradezco a mi querida y amada esposa Sandra y al fruto de nuestro amor mi hija Naomi, que siempre me apoyaron y brindaron todo su Amor.

A mis hermanas por ser parte importante de mi vida y representar la unidad familiar.

A mis sobrinos Michael, Nuria y mi cuñado Walter, gracias por su apoyo y comprensión.

A mis amigos por todos esos momentos que pasamos juntos. Por las tareas que juntos realizamos y por todas las veces que a mí me explicaron gracias. Por la confianza que en mí depositaron.

También me gustaría agradecer a mis profesores durante toda mi carrera profesional porque todos han aportado con un granito de arena a mi formación.

Son muchas las personas que han formado parte de mi vida profesional a las que les encantaría agradecerles su amistad, consejos, apoyo, ánimo y compañía en los momentos más difíciles de mi vida. Algunas están aquí conmigo y otras en mis recuerdos y en mi corazón, sin importar en donde estén quiero darles las gracias por formar parte de mí, por todo lo que me han brindado y por todas sus bendiciones.

Para ellos: Muchas gracias y que Dios los bendiga.

AGRADECIMIENTO

Luis Orlando Saquinga Sailema

Agradezco principalmente a ti Dios por bendecirme siempre para llegar hasta donde he llegado, porque hiciste realidad este sueño anhelado.

A mis padres que han sido un pilar fundamental en mi carrera quienes con su apoyo han logrado que termine exitosamente este periodo de mi vida, sin ellos nada de esto fuera posible. Además a mi hermana de la que he recibido ánimo y apoyo para lograr lo que me he propuesto, mi familia ha sido mi apoyo y hacia los cuales expreso mis más grandes agradecimientos.

A mi compañero de tesis por su amistad con quien he compartido esta gran experiencia la cual agradezco todo el apoyo y comprensión.

A todos mis profesores, amigos y amigas que siempre supieron aportar con ideas, consejos, experiencias, para así poder alcanzar mí objetivo y finalmente agradezco a todas las personas que de una u otra hicieron posible el desarrollo y la culminación exitosa de nuestro proyecto de graduación.

RESUMEN

El proyecto trata de un diseño y construcción de un sistema de control, monitoreo y vigilancia por cámara, vía internet, para ser utilizado ya sea en un domicilio u departamento, funcionan las diferentes ventajas que nos facilita la tecnología del Internet. El sistema tiene instalado un software libre, dispositivos controladores que son electroválvula y motor, una mini computadora llamada Raspberry Pi, una tarjeta de adquisición de datos llamado Arduino Uno, sensores y dos cámaras web.

El usuario tendrá que tener instalado TeamViewer en la computadora que vaya a realizar la conexión remota con la computadora que estará en conexión de área local con la mini computadora llamada Raspberry, donde podrá observar mediante imágenes establecidas por cámaras, la ubicación de su mascota y al mismo tiempo real podrá controlar los dispositivos (Electroválvula y motor) para agregar al recipiente la cantidad de alimentación y agua que desee, podrá acceder a un dispositivo de voz para llamar a su mascota en caso de no aparecer en la mira de las cámaras y podrá constatar si se está llenando los recipientes con los alimentos, mediante los sensores que están situados en los recipientes y estos llevaran un registro de consumo y verificación de cuantas veces se ha dado de alimentar al animal.

El cliente tendrá en su domicilio acceso a Internet en su computador y posteriormente tendrá una conexión en red con Raspberry Pi y este tendrá instalado el software necesario para la aplicación ya que estos estarán en conexión con los dispositivos controladores, sensores, parlante y cámaras que estarán en un soporte metálico apropiado para los dispositivos y la mascota, la primera cámara está ubicada en el soporte para observar si la mascota se está alimentando y la segunda cámara está ubicada en una zona alta de la estructura para la vigilancia de la mascota.

Todos los dispositivos del domicilio a funcionar estarán conectados a la red eléctrica pública, posteriormente estará todo esto conectado a un suministro de Almacenamiento de energía.

ABSTRACT

The project is a design and construction of a control system and camera surveillance monitoring via the Internet, for use either in a home or apartment, work the different advantages that technology facilitates Internet. The system is installed free software, devices and drivers that are solenoid engine, a Raspberry Pi mini computer called a data acquisition board called Arduino Uno, sensors and two webcams.

The user will need to have installed TeamViewer on the computer to be remotely connect to the computer that will be in the local area connection to the mini computer called Raspberry , where you can see through images provided by cameras, the location of your pet and to Real while you control devices (solenoid and motor) to add the recipient the amount of food and water you want , you can access a voice device to call your pet in case of not appearing in sight of cameras and may determine whether it is filling the containers with food , through sensors located in the containers and they keep a record of consumption and check on how often to feed the animal die .

The customer shall at his home Internet access on your computer and then have a network connection with Raspberry Pi and it will have installed the necessary software for application as these are in connection with controller devices, sensors, and cameras will be speaking in a metal support devices and appropriate for the pet, the first camera is located on the stand to see if the pet is being fed and the second chamber is located in a high area of the structure to monitor the pet.

All devices in the home to work will be connected to the public grid, then all this will be connected to a power supply Storage.

ÍNDICE GENERAL

| | |
|-------------------------------------|------|
| CARATULA..... | i |
| DECLARACION DE RESPONSABILIDAD..... | ii |
| CERTIFICADO..... | iii |
| AUTORIZACION DE PUBLICCION..... | iv |
| DEDICATORIA 1..... | v |
| DEDICATORIA 2..... | vi |
| AGRADECIMIENTO 1..... | vii |
| AGRADECIMIENTOS 2..... | viii |
| RESUMEN..... | ix |
| ABSTRACT..... | x |
| ÍNDICE DE CONTENIDOS..... | xii |
| ÍNDICE DE TABLAS..... | xvi |
| ÍNDICE DE GRÁFICOS..... | xvii |
| ÍNDICE DE ANEXOS..... | xxi |

ÍNDICE DE CONTENIDO

| | |
|--------------------------------------------------------------------------------------------------------------|----|
| CAPÍTULO 1 | 1 |
| FUNDAMENTOS TEORICOS | 1 |
| 1.1 Introducción. | 1 |
| 1.2 GENERALIDADES DE UN SISTEMA DE VIGILANCIA | 2 |
| 1.2.1 Los Servicios de un sistema de vigilancia. | 2 |
| 1.2.2 Gestión local y remota. | 3 |
| 1.2.3 Tipos de Arquitectura..... | 3 |
| 1.2.3.1 Diferencias conceptuales y operacionales entre una arquitectura centralizada y distributiva. | 4 |
| 1.2.4 SISTEMAS DE VIDEO VIGILANCIA | 5 |
| 1.2.4.1 Descripción de la Vigilancia | 5 |
| 1.2.4.2 La evolución de los sistemas de video vigilancia por video | 5 |
| 1.3 ACTUADORES, TRANSDUCTORES, SENSORES, CAMARAS..... | 6 |
| 1.3.1 Actuadores | 6 |
| 1.3.2 Transductores | 8 |
| 1.3.3 Sensores | 10 |
| 1.3.3.1 Galgas Extensiométricas | 10 |
| 1.3.4 Cámaras..... | 12 |
| 1.4 MEDIOS DE TRANSMISIÓN..... | 13 |
| 1.4.1 Modos de Transmisión (Guiados y no Guiados) | 13 |
| 1.4.1.1 Guiados: | 14 |
| 1.4.1.2 No guiados: | 14 |
| 1.4.2 Tipos de Transmisión..... | 14 |
| 1.5 RASPBERRY PI, PLACA ARDUINO UNO..... | 18 |
| 1.5.1 Raspberry pi | 18 |
| 1.5.1.1 Historia..... | 18 |
| 1.5.1.2 Introducción | 18 |
| 1.5.1.3 Hardware | 20 |
| 1.5.1.4 Software | 21 |

| | | |
|---------|-----------------------------------------------------------|----|
| 1.5.1.5 | Sistemas operativos..... | 22 |
| 1.5.1.6 | Distribución Ligeras Multipropósitos | 23 |
| 1.5.1.7 | Distribuciones ligeras de único propósito | 23 |
| 1.5.1.8 | Especificaciones técnicas | 24 |
| 1.5.2 | ARDUINO | 25 |
| 1.5.2.1 | Descripción | 26 |
| 1.5.2.2 | El hardware | 27 |
| 1.5.2.3 | Principales partes que forman una Placa Arduino Uno | 29 |
| 1.5.2.4 | Ventajas y Desventajas del Arduino | 32 |
| 1.5.2.5 | Características Principales de la placa Arduino Uno | 33 |
| 1.6 | COMUNICACIÓN DEL PC CON ELEMENTOS EXTERNOS | 33 |
| 1.6.1 | Puerto Serie | 33 |
| 1.6.2 | Puerto Paralelo | 34 |
| 1.6.3 | Puerto USB | 35 |
| 1.6.3.1 | Características del puerto USB | 35 |
| 1.7 | DISEÑO Y PUBLICACION DE LA PÁGINA WEB..... | 36 |
| 1.7.1 | El Internet..... | 36 |
| 1.7.2 | Como conectarse a Internet..... | 37 |
| 1.7.3 | Servicios Web | 37 |
| 1.7.4 | Transmisión de datos por internet..... | 38 |
| 1.7.4.1 | Modelo OSI..... | 39 |
| 1.7.4.2 | Arquitectura cliente servidor..... | 40 |
| 1.7.4.3 | Protocolo TCP/IP | 41 |
| 1.7.4.4 | IP (Internet Protocolo) | 43 |
| 1.7.4.5 | Direcciones Internet (IP) | 45 |
| 1.7.5 | Medios de acceso a internet | 48 |
| 1.7.5.1 | Para clientes domésticos | 48 |
| 1.7.5.2 | Para clientes comerciales e industriales | 50 |

| | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| CAPÍTULO 2 | 52 |
| DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE CONTROL, MONITOREO Y VIGILANCIA POR CÁMARA, VIA INTERNET, PARA LA ALIMENTACIÓN DE ANIMALES DOMESTICOS. | 52 |
| 2.1 Especificación del Diseño (Introducción)..... | 52 |
| 2.2 Análisis de la Estructura | 52 |
| 2.2.1 Operación de la Estructura..... | 54 |
| 2.2.2 Métodos de control de la Estructura. | 54 |
| 2.2.2.1 Método de forma remota desde el propio Raspberry Pi..... | 54 |
| 2.2.2.2 Método de control mediante el Internet | 55 |
| 2.3 Diseño del SOFTWARE..... | 55 |
| 2.3.1 Implementación del Software | 55 |
| 2.3.1.1 Programación en Raspberry Pi con el Software PYTHON | 55 |
| 2.3.1.2 Programación en Arduino Uno | 57 |
| 2.3.1.3 Panel Frontal Principal..... | 58 |
| 2.3.2 Programación en Raspberry Pi con el Software HTML Y PHP5 para las dos cámaras..... | 62 |
| 2.3.2.1 Ahora ya podemos acceder a ver las Webcams desde la dirección indicada | 66 |
| 2.3.2.2 Automatización de arranque y parada de motion..... | 66 |
| 2.3.2.3 Ahora realizaremos la unión de las dos cámaras | 67 |
| 2.4 Diseño del HARDWARE | 68 |
| 2.4.1 Implementación del Hardware | 68 |
| 2.4.1.1 Internet Por Modem 3g USB De Claro | 68 |
| 2.4.1.2 Router..... | 69 |
| 2.4.1.3 La Computadora..... | 69 |
| 2.4.1.4 Raspberry Pi..... | 69 |
| 2.4.1.5 Módulo I/O..... | 70 |
| 2.4.1.5.1 Hardware Del Arduino Uno..... | 70 |
| 2.4.1.5.2 Referencia para pines analógicos (AREF) | 71 |
| 2.4.1.6 Caja De Relés..... | 72 |

| | |
|----------------------------------------------------------------------------|-----|
| 2.4.1.7 Sensores Y Carga | 73 |
| 2.4.1.8 Dispositivo De Audio..... | 74 |
| 2.4.1.8.1 Decodificador Wtv020-Sd-16p..... | 74 |
| 2.5 Servicio de Internet proporcionado por distribuidor de Internet..... | 76 |
| 2.6 Integración total del sistema de control, monitoreo y vigilancia. | 81 |
| 2.6.1 Diagrama de Bloques del sistema | 81 |
| 2.6.2 Diagrama total del circuito de control..... | 82 |
| | |
| CAPÍTULO 3 | 84 |
| PRUEBAS Y ANÁLISIS DE RESULTADOS | 84 |
| 3.1 Descripción física del diseño | 84 |
| 3.2 PRUEBAS EXPERIMENTALES | 84 |
| 3.3 Pruebas del sistema HTML. | 85 |
| 3.4 Análisis de resultados..... | 86 |
| 3.5 Análisis Técnico - Económico | 89 |
| 3.6 ALCANCES Y LIMITACIONES | 91 |
| 3.6.1 Alcances | 91 |
| 3.6.2 Limitaciones..... | 91 |
| | |
| CAPITULO 4 | 93 |
| CONCLUSIONES Y RECOMENDACIONES..... | 93 |
| 4.1 CONCLUSIONES | 93 |
| 4.2 RECOMENDACIONES..... | 95 |
| BIBLIOGRAFÍA | 97 |
| ANEXOS | 102 |
| GLOSARIO DE TERMINOS..... | 168 |

ÍNDICE DE TABLAS

CAPITULO I

| | |
|----------------------------------------------------------------------|----|
| Tabla 1.1 Diferencias de tipos de arquitecturas | 4 |
| Tabla 1.10 Direcciones IP de Internet..... | 42 |
| Tabla 1.11 Representación decimal de direcciones de Internet | 43 |
| Tabla 1.2 Evolución de un sistema de vigilancia..... | 6 |
| Tabla 1.3 Ventajas y desventajas de cada tipo de actuador: | 7 |
| Tabla 1.4 Clasificación de sensores | 8 |
| Tabla 1.5 Conexiones estándar de un cable cruzado..... | 16 |
| Tabla 1.6 Características principales de la placa Raspberry PI. | 18 |
| Tabla 1.7 Especificaciones técnicas de la placa Raspberry pi | 22 |
| Tabla 1.8 Pines de una comunicación en serie..... | 31 |
| Tabla 1.9 Niveles del modelo OSI..... | 36 |

CAPITULO III

| | |
|----------------------------------------------------------------------|----|
| Tabla 3 Ración para la alimentación Diaria de un perro..... | 82 |
| Tabla 3.2 Pruebas en la electroválvula..... | 83 |
| Tabla 3.3 Valores de peso del Bebedero del Animal. | 83 |
| Tabla 3.4 Pruebas en la Banda Transportadora..... | 83 |
| Tabla 3.5 Valores de peso de la Alimentación sólida del Animal. | 84 |

ÍNDICE DE GRAFICOS

CAPITULO I

| | |
|----------------------------------------------------------------------------------------------------------------------|----|
| Figura 1.1 Monitoreo y Control a través de Internet..... | 2 |
| Figura 1.2. Tipos de Arquitecturas..... | 4 |
| Figura 1.3 Galgas extensiométricas..... | 11 |
| Figura 1.4 Cámara Web Vx800 Microsoft..... | 12 |
| Figura 1.5 Forma de conexión de un cable directo..... | 16 |
| Figura 1.6 Forma de conexión de un cable cruzado..... | 17 |
| Figura 1.7 Placa frontal del Raspberry PI..... | 18 |
| Figura 1. 8 Diagrama que muestra la localización de cada uno de los componentes de la placa Raspberry Pi..... | 19 |
| Figura 1.9 Diagrama de bloques del modelo B, en el Modelo A, el puerto USB está conectado directamente a SoC..... | 21 |
| Figura 1.10 Placa Arduino Uno..... | 25 |
| Figura 1. 11 Partes constitutivas de la placa Arduino..... | 28 |
| Figura 1.12 Conexión del puerto paralelo..... | 33 |
| Figura 1.13 Tipos y polaridad de los puertos USB..... | 33 |
| Figura 1.14 Enlace de datos a través del Internet..... | 35 |
| Figura 1.15 Conexión cliente servidor por medio de Internet..... | 36 |
| Figura 1.16 Enlace del modelo OSI..... | 37 |
| Figura 1.17 Arquitectura cliente servidor..... | 38 |
| Figura 1.18 Comparación de capas del modelo OSI y TCP/IP..... | 40 |
| Figura 1. 19 Internet por MODEM..... | 46 |
| Figura 1.20 Sistema de conexión a Internet con GPRS y GPS..... | 48 |

CAPITULO II

| | |
|--------------------------------------------------------------------------------------|----|
| Figura 2.1: Diagrama de bloques del control, monitoreo y vigilancia de canes..... | 51 |
| Figura 2.2: Etapas del hardware del sistema sin Internet..... | 52 |
| Figura 2.3: Etapas del hardware del sistema con Internet..... | 53 |
| Figura 2.4: Visualización del Escritorio del Raspberry Pi..... | 54 |

| | |
|---------------------------------------------------------------------------------------------|----|
| Figura 2.5: Ventana donde se escoge la opción Ejecutar del programa de alimentación..... | 54 |
| Figura 2.6: Ventana del proceso de alimentación..... | 55 |
| Figura 2.7: Ventana que contiene el programa realizado en Arduino..... | 55 |
| Figura 2.8: Panel frontal del proceso de alimentación..... | 56 |
| Figura 2.9: Botones en estado de Apagado de la Alimentación Sólida..... | 57 |
| Figura 2.10: Botones en estado de Encendido de la Alimentación Sólida..... | 57 |
| Figura 2.11: Botones en estado de Apagado de la Alimentación Líquida..... | 57 |
| Figura 2.12: Botones en estado de Encendido de la Alimentación Líquida..... | 58 |
| Figura 2.13: Botones en estado de Apagado de la Luz de los Alimentos..... | 58 |
| Figura 2.14: Botones en estado de Encendido de la Luz de los Alimentos..... | 58 |
| Figura 2.15: Botones en estado de Apagado de la Luz de Exteriores..... | 59 |
| Figura 2.16: Botones en estado de Encendido de la Luz de Exteriores..... | 59 |
| Figura 2.17: Botones de audio..... | 59 |
| Figura 2.18: Botones de registro de alimentos..... | 60 |
| Figura 2.19: Ventana donde se encuentran los registros de consumo de los alimentos..... | 60 |
| Figura 2.20: Ventana para instalar en programa motion de las cámaras..... | 60 |
| Figura 2.21: Ventana de configuración de motion..... | 61 |
| Figura 2.22: Ventana de configuración de motion.conf..... | 62 |
| Figura 2.23: Ventana de configuración para la cámara 1..... | 62 |
| Figura 2.24: Ventana de configuración para la cámara 2..... | 63 |
| Figura 2.25: Ventana de verificación de instalación de las dos cámaras cam1 y cam2..... | 63 |
| Figura 2.26: Ventana de configuración de motion el auto inicio de las cámaras..... | 64 |
| Figura 2.27: Cámaras establecidas en su dirección adecuada..... | 64 |
| Figura 2.28: Ventana donde se muestra las dos cámaras con su dirección establecida..... | 65 |
| Figura 2.29: Estructura frontal del sistema de alimentación..... | 66 |
| Figura 2.30: Forma Física del Arduino uno..... | 68 |

| | |
|------------------------------------------------------------------------------------------------------------------------|----|
| Figura 2.31: Arduino uno y sus elementos más importantes de su placa..... | 69 |
| Figura 2.32: Diseño de la entrada y salida del relé para el manejo del actuador. | 70 |
| Figura 2.33: Forma física los cuatro relés que se conectaran a los actuadores..... | 70 |
| Figura 2.34: Fuente de alimentación para los módulos de carga..... | 71 |
| Figura 2.35: Sensor y actuador del Proceso..... | 71 |
| Figura 2.36: Acondicionamiento de señal del Sensor (Galga extensiométrico)..... | 72 |
| Figura 2.37: Panel frontal del dispositivo de reproducción y su distribución de pines con su tarjeta SD..... | 73 |
| Figura 2.38: Descripción de pines del módulo de reproducción de audio WTV020-SD-16P..... | 73 |
| Figura 2.39: Conexiones para la reproducción de audio..... | 74 |
| Figura 2.40: Ventana del programa TeamViewer..... | 74 |
| Figura 2.41: Corrido del programa TeamViewer en Escritorio Remoto..... | 75 |
| Figura 2.42: Corrido del programa Xming para acceder al Escritorio Remoto del Raspberry Pi..... | 75 |
| Figura 2.43: Escritorio Remoto del Raspberry Pi corrido desde Xming..... | 76 |
| Figura 2.44: Visualización de las cámaras por Internet por medio de Acceso Remoto..... | 76 |
| Figura 2.45: Escritorio del Raspberry donde correremos el Programa Alimentacióncan.py..... | 77 |
| Figura 2.46: Ventana donde se escoge la opción Ejecutar..... | 77 |
| Figura 2.47: Ventana del proceso de Alimentación, observado desde el Internet por Acceso Remoto con TeamViewer..... | 78 |
| Figura 2.49 Diagrama en bloques de los elementos del sistema..... | 79 |
| Figura 2.50 Diagrama Total del circuito de control..... | 80 |
| Figura 2.51 Conexión total del sistema eléctrico de la alimentación..... | 80 |
| Figura 2.52 En las figuras a y b se indican la cámara y una bombilla de exteriores para la vigilancia del can..... | 81 |

Figura 2.53 Sistema de alimentación para canes totalmente terminada.....81

CAPITULO III

Figura 3.1 Verificación de datos guardados
de la Alimentación Sólida y Líquida.....87

Figura 3.2 Verificación de datos guardados al presionar el botón de la
Alimentación Sólida y Líquida.....88

ANEXOS

ANEXO A

Instalación y Configuraciones de la mini computadora RASPBERRY PI.....99

ANEXO B

Instalación y Configuraciones de la Tarjeta Electrónica.

Programable Arduino Uno.....108

ANEXO C

Instalación del TeamViewer.....111

ANEXO D

Desarrollo del programa Python Idle.....116

ANEXO E

Programación de la Tarjeta Electrónica Arduino.....127

ANEXO F

Código d Programación de las dos cámaras.....132

ANEXO G

Manual de usuarios para la alimentación del animal.....136

ANEXO H

Manual de operador.....149

ANEXO I

Instalación y Configuración Xming.....156

CAPÍTULO 1

FUNDAMENTOS TEORICOS

1.1 Introducción.

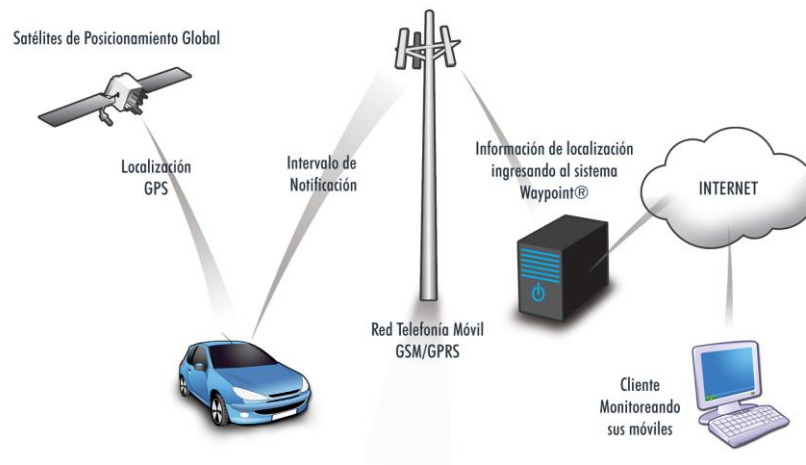
El control de procesos mediante Internet ha alcanzado en la actualidad un amplio interés en diferentes áreas de la industria. Actualmente es posible obtener resultados a distancia de alta calidad aprovechando la utilidad que brindan el Internet y el uso de las computadoras (PC).

Generalmente el monitoreo a distancia se realiza con una estructura cliente – servidor en la que ambos deben poseer una aplicación en cada extremo de la conexión y así compartir información.

El Sistema de Monitoreo y Control mediante Internet (Figura1.1) puede ser manejado desde cualquier PC conectada a Internet y que cuente con un programa navegador. Dicha computadora (cliente) no requiere instalación alguna de controladores o software específico para el manejo del Sistema.

Mediante este Sistema es posible monitorear y controlar procesos industriales. Es absolutamente configurable desde el cliente, respondiendo en forma automática a los cambios en el proceso, según los requerimientos configurados por el usuario.

Las operaciones que controlan el Sistema son realizadas en el equipo Servidor, el cual posee un Sistema Operativo de servidor y utiliza la tecnología ASP que le proporciona características de Sitio Web dinámico, ya que la información enviada por el cliente a través del explorador es interpretada y ejecutada en el Servidor, que realizará las acciones correspondientes sobre el Sistema, enviando finalmente los resultados al explorador del cliente.



¹Figura1.1 Monitoreo y Control a través de Internet.

1.2 GENERALIDADES DE UN SISTEMA DE VIGILANCIA

1.2.1 Los Servicios de un sistema de vigilancia.

El sistema de video vigilancia facilita una infraestructura para nuevos servicios muy interesantes para usuarios y proveedores de diferentes aplicaciones de internet.

Este sistema permite acceder al tiempo real en cualquier momento desde tu celular o cualquier ordenador, este donde este. El video se almacena en ubicaciones remotas, por motivos de comodidad o seguridad y la información se transmitirá a través de la red LAN o Internet. Esto significa que tanto las casas e incluso las empresas, establecimientos pequeños y dispersos pueden hacer un uso eficaz de la solución de vigilancia IP en aplicaciones de seguridad o supervisión a distancia.

Este sistema de vigilancia privada le permite el monitoreo y control de un sistema de alimentación de animales domésticos mediante el uso del internet como medio primordial de la comunicación.

Existen, y se están desarrollando, un gran numero e servicios que se puede realizar o crear gracias a los sistemas de video vigilancia, para con ello mejorar la

¹ https://www.waypoint.cl/layout.php?a=servicio_flotagps

vida cotidiana de los usuarios y tener una comunicación continúa con sus animales de hogar (canes).

El uso de un sistema de vigilancia para supervisar el lugar donde vivimos durante nuestra ausencia, una posible solución es invertir en cámaras de circuito cerrado, un ordenador casero basado en sistema de video vigilancia, es una herramienta eficaz que cualquier persona con un ordenador y cámaras web puede instalar.

Una PC basado en sistema de video vigilancia es también ideal para el seguimiento de sus mascotas en el hogar, o la puerta de entrada, garaje y cuartos mientras que usted está ausente, en el trabajo o de vacaciones.

1.2.2 Gestión local y remota.

Gracias a la integración de sistema de video vigilancia en el hogar se puede hacer una gestión mejor y más personalizada del mismo. Esto permite ahorrar dinero, tener mejor confort y mejorar la seguridad, como por ejemplo: la programación controlada del sistema de alimentación para nuestros canes y con esto lograremos mejoras su estilo de vida y costumbres de la simulación de presencia de todos los Habitantes de la familia, al momento de establecer una comunicación con el can, aunque ellos se encuentren fuera de las instalaciones durante el día o cuando estén en vacaciones. Las cámaras web permitirán a los habitantes observar y vigilar a sus mascotas que se encuentren en perfecto estado de alimentación, caso contrario realizar el control de nutrición respectivo. Medición de consumo de alimentación, a través de sensores que le permitirán administrar la cantidad adecuada de alimentación sin desperdiciarla.

1.2.3 Tipos de Arquitectura²

Arquitectura Centralizada: un controlador centralizado recibe información de múltiples sensores y, una vez procesada, genera las órdenes oportunas para los actuadores.

³Arquitectura Distribuida: en este caso, no existe la figura del controlador centralizado, sino que toda la inteligencia del sistema está distribuida por todos

² <http://www-oei.eui.upm.es/Asignaturas/BD/BD/docbd/tema/Arquitectura.pdf>

³ <http://www.slideshare.net/jpbthames/arquitectura-de-sistemas-distribuidos>

los módulos sean sensores o actuadores. Suele ser típicos de los sistemas de cableado en bus.

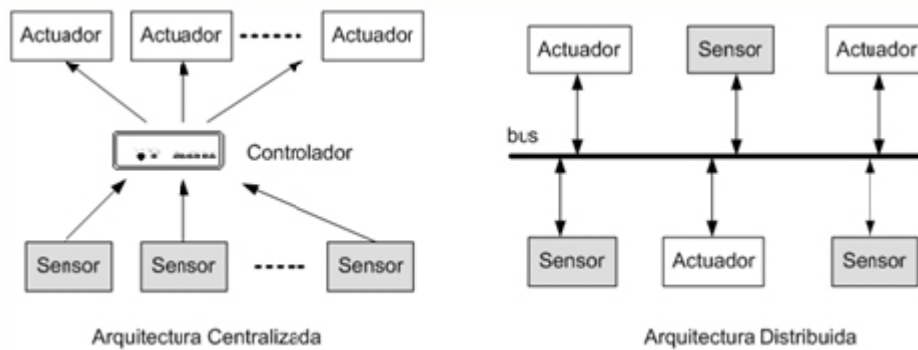


Figura 1.2. Tipos de Arquitecturas

1.2.3.1 Diferencias conceptuales y operacionales entre una arquitectura centralizada y distributiva⁴.

Tabla 1.1 Diferencias de tipos de arquitecturas.

| ASPECTO | SCADA | DCS |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| TIPO DE ARQUITECTURA | CENTRALIZADA | DISTRIBUIDA |
| TIPO DE CONTROL PREDOMINANTE | SUPERVICION: Supervisión y monitoreo a lazo cerrado. No es aconsejables lazos cerrados de control Adicionalmente: control secuencial regulatorio. | REGULATORIOS: Lazos de control cerrados automáticamente por el sistema. Adicionalmente: control secuencial, batch, algoritmos avanzados, etc. |
| TIPOS DE VARIABLES | DESACOPLADAS | ACOPLADAS |
| UNIDADES DE ADQUISICIÓN DE DATOS Y | Remotas, PLCs. | Control de lazo, Plcs. |

⁴ <http://www.raulcarretero.com/2012/04/17/por-que-y-cuando-elegir-un-sistema-domotico-centralizado-o-distribuido/>

| | | |
|------------------------|-----------------------------------------------------------------|----------------------------------------|
| CONTROL | | |
| MEDIOS DE COMUNICACIÓN | Radio, satélite, líneas telefónicas, conexión directa, LAN, WAN | Redes de área local, conexión directa. |
| BASE D DATOS | CENTRALIZADA | DISTRIBUIDA |

1.2.4 SISTEMAS DE VIDEO VIGILANCIA

Los sistemas de vigilancia por video existen a partir de los años 70. Empezaron siendo sistemas totalmente analógicos desde la aparición de las primeras cámaras analógicas con tubo conectado a Vcr2 y paulatinamente se fueron digitalizando. Los sistemas de hoy utilizan cámaras y servidores de PC para la grabación de video pero no constituyen sistemas completamente digital por diversas soluciones que son parcialmente digitales.

En Video Vigilancia disponemos de la mayor gama de soluciones profesionales de video para vigilar y controlar su negocio, domicilio y seres queridos, tanto local como remotamente desde internet. Los sistemas de video vigilancia le permiten ver las cámaras de su negocio en cualquier momento y desde cualquier parte del mundo con solo una conexión de internet.

1.2.4.1 Descripción de la Vigilancia.

El sistema de video vigilancia consiste en instalar cámaras de video que son grabadas en un grabador digital y que pueden ser vistas en un monitor central. Los sistemas de video vigilancia son muy sencillos de utilizar ya que se manejan de forma similar a un video doméstico.

1.2.4.2 La evolución de los sistemas de video vigilancia por video⁵

Los sistemas de vigilancia por vídeo existen desde hace 25 años. Empezaron siendo sistemas analógicos al 100% y paulatinamente se fueron digitalizando.

⁵ http://www.axis.com/es/products/video/about_networkvideo/evolution.htm

Los sistemas de hoy en día han avanzado mucho desde la aparición de las primeras cámaras analógicas con tubo conectadas a VCR.

En la actualidad, estos sistemas utilizan cámaras y servidores de PC para la grabación de vídeo en un sistema completamente digitalizado. Sin embargo, entre los sistemas completamente analógicos y los sistemas completamente digitales existen diversas soluciones que son parcialmente digitales. Dichas soluciones incluyen un número de componentes digitales pero no constituyen sistemas completamente digitales.

Tabla 1.2 Evolución de un sistema de vigilancia

| | |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Fully analog: | Sistemas de circuitos cerrados de TV analógicos usando VCR. |
| Partly digital: | Sistemas de circuito cerrado de TV analógicos usando DVR. Sistemas de circuito cerrado de TV analógicos usando DVR de red. |
| Fully digital: | Sistemas de vídeo IP que utilizan servidores de vídeo. Network video systems using network cameras. |

1.3 ACTUADORES, TRANSDUCTORES, SENSORES, CAMARAS

1.3.1 Actuadores⁶

Un actuador es un dispositivo capaz de transformar energía hidráulica, neumática o eléctrica en la activación de un proceso con la finalidad de generar un efecto sobre un proceso automatizado. Este recibe la orden de un regulador o controlador y en función a ella genera la orden para activar un elemento final de control como, por ejemplo, una válvula.

⁷El actuador más común es el actuador manual o humano. Es decir, una persona mueve o actúa un dispositivo para mover su funcionamiento.

⁶ <http://es.wikipedia.org/wiki/Actuador>

⁷ <http://www.monografias.com/trabajos15/actuadores/actuadores.shtml>

Con el tiempo, se hizo conveniente automatizar la actuación de dispositivos, por lo que diferentes dispositivos hicieron su aparición. Actualmente hay básicamente dos tipos de actuadores:

- Lineales
- Rotativos

Los actuadores lineales generan una fuerza en línea recta, tal como haría un pistón. Los actuadores rotativos generan una fuerza rotativa, como lo haría un motor eléctrico.

Existen varios tipos de actuadores como son:

- Electrónicos
- Hidráulicos
- Neumáticos
- Eléctricos.

Los actuadores Hidráulicos, neumáticos y eléctricos son usados para manejar aparatos mecatrónicos. Por lo general, los actuadores hidráulicos se emplean básicamente cuando lo que se necesita es potencia, y los neumáticos son simples posicionamientos. Sin embargo, los actuadores hidráulicos requieren mucho equipo para el suministro de energía así como el mantenimiento periódico. Por otro lado las aplicaciones de los modelos neumáticos también son limitadas desde el punto de vista de precisión y mantenimiento.

⁸Tabla 1.3 Ventajas y desventajas de cada tipo de actuador:

| TIPO DE ACTUADOR | VENTAJAS | DESVENTAJAS |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| Neumático | <ul style="list-style-type: none"> • Bajo costo • Rápido • Sencillo • Robusto | <ul style="list-style-type: none"> • Requiere de instalaciones especiales. • Son muy Ruidosos. |

⁸ <http://solorobotica.blogspot.com/2011/08/actuadores-en-robotica.html>

| | | |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Hidráulicos | <ul style="list-style-type: none"> • Rápido • Alta capacidad de carga. • Presentan estabilidad frente a cargas estáticas. | <ul style="list-style-type: none"> • Requieren instalaciones especiales. • Son de difícil mantenimiento. • Resultan poco económicos. |
| Eléctricos | <ul style="list-style-type: none"> • Precisos y fiables • Silenciosos • Su control es sencillo. • Son de fácil instalación. | <ul style="list-style-type: none"> • Su potencia es limitada. |

1.3.2 Transductores

Un transductor es un dispositivo capaz de transformar o convertir un determinado tipo de energía de entrada, en otra diferente a la salida.

El nombre del transductor ya nos indica cual es la transformación que realiza (por ejemplo electromecánica, transforma una señal eléctrica en mecánica o viceversa).

Es un dispositivo usado principalmente en la industria, en la medicina, en la agricultura, en robótica, en aeronáutica, etc. para obtener la información de entornos físicos y químicos y conseguir (a partir de esta información) señales o impulsos eléctricos o viceversa. Los transductores siempre consumen algo de energía por lo que la señal medida resulta debilitada.

Por lo tanto, la función lineal, por su sencillez, será un atributo deseable en un transductor. A esta característica habrá que las siguientes:

1. Sensibilidad del transductor: de qué tamaño es salida en relación con la entrada.
2. Estabilidad: qué tanto se modifica la con el cambio de variables ajenas a la entrada (incluyendo el tiempo).
3. Complejidad en su condicionamiento: qué tan complejo es el circuito que se debe poner en su salida para manejarla.

En esta sección se presentará una introducción al principio de funcionamiento y forma de empleo de los transductores más comúnmente empleados en instrumentación. Los tipos de transductores más comunes son:

- Temperatura: termopar, termistor y estado sólido.
- Desplazamiento.
- Presión.
- Magnetismo.
- Velocidad.
- Radiación.
- Esfuerzo y Deformación
- Fuerza
- Velocidad
- Flujo

⁹Tabla 1.4 Clasificación de sensores de acuerdo a su variable medida, a él transductor usado y al tipo de señal de salida.

| MAGNITUD | TRANSDUCTOR | CARACTERISTICA |
|---------------------------------------|---------------------------|-----------------------|
| Posición lineal o angular | Potenciómetro | Analógica |
| | Encoger | Digital |
| Desplazamiento y Deformación | Transformador diferencial | Analógica |
| | Galga extensiométrica | Analógica |
| Velocidad lineales y angulares | Dinamo tacométrica | Analógica |
| | Encoder | Digital |
| | Detector inductivo | Digital |
| Aceleración | Acelerómetro | Analógico |
| | Sensor de velocidad | Digital |
| Fuerza y par | Galgas | Analógicas |
| Presión | Membranas | Analógicas |

| | | |
|------------------------------|--------------------|-----------------------|
| | Piezoeléctricos | Analógicas |
| Caudal | Turbina | Analógica |
| | Magnético | Analógica |
| Temperatura | Termopar | Analógica |
| | PT100 | Analógica |
| | NTC | Analógica |
| | PTC | I/O |
| | Bimetal | I/O |
| Sensores de presencia | Inductivos | I/O |
| | Capacitivos | I/O |
| | Ópticos | I/O Analógicas |
| Sensores Táctiles | Matriz de contacto | I/O |
| | Piel artificial | Analógica |
| Visión Artificial | Cámara de video | Procesamiento digital |
| | Cámaras CCD | Procesamiento Digital |

1.3.3 Sensores

1.3.3.1 ¹⁰Galgas Extensiométricas

Una galga extensiométrica es un sensor basado en el efecto piezorresistivo. Un esfuerzo que deforma la galga extensiométrica, generará una variación en su resistencia eléctrica

Las galgas extensiométricas se basan en aprovechar la resistencia eléctrica del conductor, que al deformarse, se vuelve más estrecho y alargado, incrementando su resistencia eléctrica. Se aprovecha por tanto esta deformación para transformarlo en una señal eléctrica que permite realizar mediciones en la deformación de objetos.

¹⁰ <http://www.slideshare.net/veronik211/clasificacin-de-los-transductores>

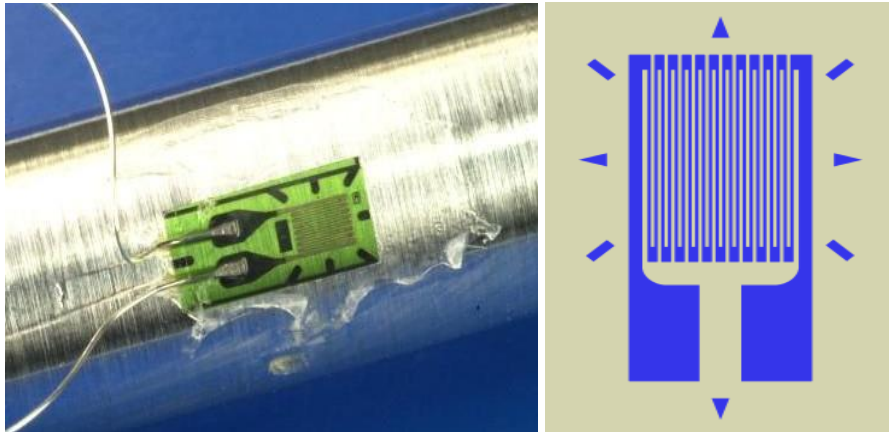


Figura 1.3 Galgas extensiométricas

La resistencia eléctrica de la galga metálica está dada por la relación entre la resistividad y la longitud respecto al área transversal.

Pueden ser:

- Hilo Metálico: Están adheridas a una base con medidas constantes, estas presentan errores cuando existen estados tensionales y son las más sencillas. Están compuestas por una película de protección, un soporte, un hilo de medida y las terminales de conexión.
- Película Metálica: Esta clase de galgas tiene una característica de fabricación similar a los circuitos impresos que tienen bases flexibles. Se desarrollan por el medio de creación de placas utilizando fotografías, llamado el método de fotograbado. Se conforman por una película de protección, un soporte, un par de conexión y de zonas anchas para reducir el efecto de tensiones transversales.
- Metal depositado: Son aplicadas directamente sobre la superficie mediante dos métodos la evaporización o el bombardeo químico.

Las principales aleaciones que usan las galgas metálicas son:

- Cobre y hierro
- Platina y silicialista
- Constantán

1.3.4 Cámaras

Una cámara web o cámara de red (en inglés: Webcams) es una pequeña cámara digital conectada a una computadora la cual puede capturar imágenes y transmitir las a través de Internet, ya sea a una página web o a otra u otras computadoras de forma privada.

Las cámaras web necesitan una computadora para transmitir las imágenes. Sin embargo, existen otras cámaras autónomas que tan sólo necesitan un punto de acceso a la red informática, bien sea Ethernet o inalámbrico. Para diferenciarlas las cámaras web se las denomina cámaras de red.



¹¹Figura 1.4 Cámara Web Vx800 Microsoft

Por lo general puede transmitir imágenes en vivo, pero también puede capturar imágenes o pequeños videos (dependiendo del programa de la cámara web) que pueden ser grabados y transmitidos por Internet. Este se clasifica como dispositivo de entrada, ya que por medio de él podemos transmitir imágenes hacia la computadora.

¹¹ <http://www.tusequipo.com/2009/10/19/microsoft-lifecam-vx-800-una-camara-web-simple-y-barata/>

1.4 ¹²MEDIOS DE TRANSMISIÓN

La comunicación es la transferencia de información de un lugar a otro, mientras que la información es un patrón físico al cual se le ha asignado un significado comúnmente acordado el patrón debe ser único -separado y distinto-, capaz de ser enviado por un transmisor y de ser detectado y entendido por un receptor.

Así, la información es transmitida a través de señales eléctricas u ópticas utilizando un canal de comunicación o medio de transmisión.

Por medio de transmisión, la aceptación amplia de la palabra, se entiende el material físico cuyas propiedades de tipo electrónico, mecánico, óptico, o de cualquier otro tipo se emplea para facilitar el transporte de información entre terminales distante geográficamente.

Los medios de transmisión son facilidades para la transmisión por cable (todas las líneas físicas) o radio de todos los tipos de mensajes o información en forma análoga o digital, incluyendo voz, datos a diferentes velocidades, facsímile, videotelefonía, videoconferencia, televisión y pantallas visuales.

Existen dos principales categorías: línea física y radio. Lo forman cables abiertos, cables de par y cables coaxiales (terrestres y submarinos), fibra óptica, mientras que la última incluye microondas, troposférica (radio HF), y comunicaciones satelitales.

Muchos factores influyen en la elección del correcto medio de transmisión. Algunos factores son obvios como lo son los económicos; otros factores son de naturaleza técnica; mientras que todavía otros factores están relacionados directamente a los servicios y al medio en los cuales los servicios van a ser provistos.

1.4.1 ¹³ Modos de Transmisión (Guiados y no Guiados)

Dentro de los medios de transmisión habrá medios guiados y medios no guiados; la diferencia radica que en los medios guiados el canal por el que se transmite las

¹² <http://www.monografias.com/trabajos37/medios-transmision/medios-transmision.shtml>

¹³ <http://www.monografias.com/trabajos37/medios-transmision/medios-transmision2.shtml>

señales son medios físicos, es decir, por medio de un cable; y en los medios no guiados no son medios físicos.

1.4.1.1 Guiados:

- **Alambre:** se usó antes de la aparición de los demás tipos de cables (surgió con el telégrafo).
- **Guía de honda:** verdaderamente no es un cable y utiliza las microondas como medio de transmisión.
- **Fibra óptica:** es el mejor medio físico disponible gracias a su velocidad y su ancho de banda, pero su inconveniente es su coste.
- **Par trenzado:** es el medio más usado debido a su comodidad de instalación y a su precio.
- **Coaxial:** fue muy utilizado pero su problema venía porque las uniones entre cables coaxial eran bastante problemáticas.

1.4.1.2 No guiados:

- **Infrarrojos:** poseen las mismas técnicas que las empleadas por la fibra óptica pero son por el aire. Son una excelente opción para las distancias cortas, hasta los 2km generalmente.
- **Microondas:** las emisiones pueden ser de forma analógica o digitales pero han de estar en la línea visible.
- **Satélite:** sus ventajas son la libertad geográfica, su alta velocidad.... pero sus desventajas tiene como gran problema el retardo de las transmisiones debido a tener que viajar grandes distancias.
- **Ondas cortas:** también llamadas radio de alta frecuencia, su ventaja es que se puede transmitir a grandes distancias con poca potencia y su desventaja es que son menos fiables que otras ondas.
- **Ondas de luz:** son las ondas que utilizan la fibra óptica para transmitir por el vidrio.

1.4.2 ¹³Tipos de Transmisión

Según el sentido de la transmisión podemos encontrarnos con tres tipos diferentes:

- **Simplex:** Este modo de transmisión permite que la información discorra en un solo sentido y de forma permanente, con esta fórmula es difícil la corrección de errores causados por deficiencias de línea. Como ejemplos de la vida diaria tenemos, la televisión y la radio.

- **Half Duplex:** En este modo, la transmisión fluye como en el anterior, o sea, en un único sentido de la transmisión de dato, pero no de una manera permanente, pues el sentido puede cambiar. Como ejemplo tenemos los Walkis Talkis.

- **Full Duplex:** Es el método de comunicación más aconsejable, puesto que en todo momento la comunicación puede ser en dos sentidos posibles y así pueden corregir los errores de manera instantánea y permanente. El ejemplo típico sería el teléfono.

- **RS-232C:** RS-232-C estándar, en informática, estándar aceptado por la industria para las conexiones de comunicaciones en serie. Adoptado por la Asociación de Industrias Eléctricas, el estándar RS-232-C recomendado (RS es acrónimo de Recommended Standard) define las líneas específicas y las características de señales que utilizan las controladoras de comunicaciones en serie. Con el fin de estandarizar la transmisión de datos en serie entre dispositivos.

Puede funcionar bien en recorridos de cable mucho más largos con todas las velocidades pero siempre habrá riesgo de pérdida de datos.

La transmisión digital es la transmisión de pulsos digitales, entre dos puntos, en un sistema de comunicación. Con los sistemas de transmisión digital, se requieren una facilidad física tal como un par de alambres metálicos, un cable coaxial o un vínculo de fibra óptica para interconectar a los dos puntos en el sistema. Los pulsos están contenidos dentro de y se propagan con la facilidad de transmisión.

- **Transmisión asíncrona:** La manera más fácil de conseguir sincronismo es enviando pequeñas cantidades de bits a la vez, sincronizándose al inicio de cada cadena. Esto tiene el inconveniente de que cuando no se transmite ningún carácter, la línea está desocupada. Para detectar errores, se utiliza un bit de paridad en cada cadena.

Este tipo de transmisión es sencilla y no costosa, aunque requiere muchos bits de comprobación y de control.

- **Transmisión síncrona:** En este tipo de transmisión no hay bits de comienzo ni de parada, por lo que se transmiten bloques de muchos bits. Para evitar errores de delimitación, se pueden sincronizar receptor y emisor mediante una línea aparte (método utilizado para líneas cortas) o incluyendo la sincronización en la propia señal (codificación Manchester o utilización de portadoras en señales analógicas). A todo el conjunto de bits y datos se le llama trama.

Para bloques grandes de datos, la transmisión síncrona es más eficiente que la asíncrona.

- **Cable RJ-45** (registered jack 45) es una interfaz física comúnmente usada para conectar redes de cableado estructurado, (categorías 4, 5, 5e, 6 y 6a). Es parte del Código Federal de Regulaciones de Estados Unidos. Posee ocho pines o conexiones eléctricas, que normalmente se usan como extremos de cables de par trenzado.

Es utilizada comúnmente con estándares como TIA/EIA-568-B, que define la disposición de los pines o wiring pinout.

Una aplicación común es su uso en cables de red Ethernet, donde suelen usarse 8 pines (4 pares). Otras aplicaciones incluyen terminaciones de teléfonos (4 pines o 2 pares) por ejemplo en Francia y Alemania, otros servicios de red como RDSI y T1 e incluso RS-232.

- ¹⁴**Cable Directo:** El cable directo de red sirve para conectar dispositivos desiguales, como un computador con un hub o switch. En este caso ambos extremos del cable deben tener la misma distribución. No existe diferencia alguna en la conectividad entre la distribución 568B y la distribución 568A siempre y cuando en ambos extremos se use la misma, en caso contrario hablamos de un cable cruzado.

El esquema más utilizado en la práctica es tener en ambos extremos la distribución 568B.

¹⁴ <http://www.monografias.com/trabajos37/medios-transmision/medios-transmision2.shtml>

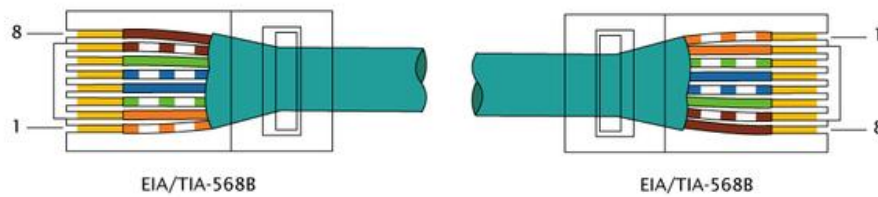


Figura 1.5 Forma de conexión de un cable directo

• **Cable Cruzado:** Un cable cruzado es un cable que interconecta todas las señales de salida en un conector con las señales de entrada en el otro conector, y viceversa; permitiendo a dos dispositivos electrónicos conectarse entre sí con una comunicación full duplex. El término se refiere - comúnmente - al cable cruzado de Ethernet, pero otros cables pueden seguir el mismo principio. También permite transmisión confiable vía una conexión Ethernet.

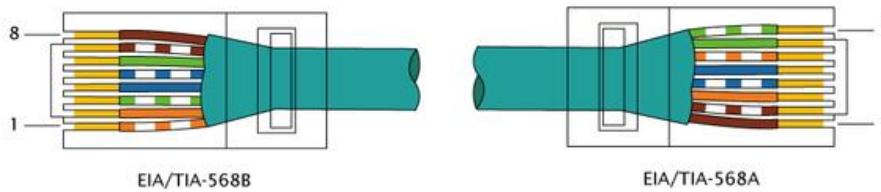


Figura 1.6 Forma de conexión de un cable cruzado

Tabla 1.5 Conexiones estándar de un cable cruzado.

| Norma A | Norma B |
|-------------------|-------------------|
| 1. Blanco/Verde | 1. Blanco/Naranja |
| 2. Verde | 2. Naranja |
| 3. Blanco/Naranja | 3. Blanco/Verde |
| 4. Azul | 4. Blanco/Azul |
| 5. Blanco/Azul | 5. Azul |
| 6. Naranja | 6. Verde |
| 7. Blanco/Marrón | 7. Blanco/Marrón |
| 8. Marrón | 8. Marrón |

1.5 RASPBERRY PI, PLACA ARDUINO UNO

1.5.1 Raspberry pi.

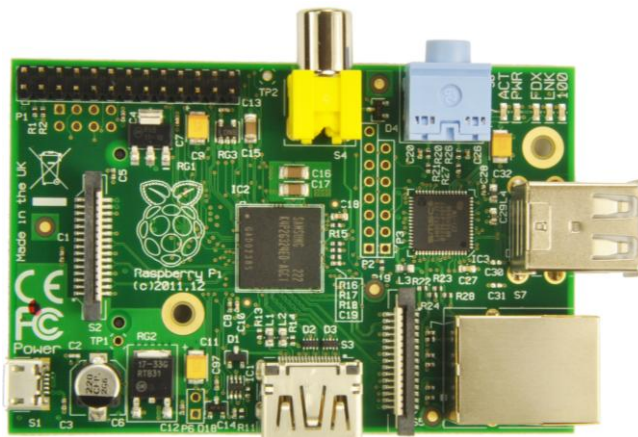
1.5.1.1 Historia

¹⁵En 2006, los primeros diseños de Raspberry Pi se basaban en el microcontrolador Atmel ATmega644. Sus esquemas y el diseño del circuito impreso están disponibles para su descarga pública.

En mayo de 2009, la Fundación Raspberry Pi fue fundada en Caldecote, South Cambridgeshire, Reino Unido como una asociación caritativa que es regulada por la Comisión de Caridad de Inglaterra y Gales.

El administrador de la fundación, Eben Upton, se puso en contacto con un grupo de profesores, académicos y entusiastas de la informática para crear un ordenador con la intención de animar a los niños a aprender informática como lo hizo en 1981 el ordenador Acorn BBC Micro. El primer prototipo basado en ARM se montó en un módulo del mismo tamaño que una memoria USB. Tenía un puerto USB en un extremo y un puerto HDMI en el otro.

1.5.1.2 Introducción



¹⁶Figura 1.7 Placa frontal del Raspberry PI.

¹⁵ http://es.wikipedia.org/wiki/Raspberry_Pi

¹⁶ <http://www.raspberrypi.org/phpBB3/viewtopic.php?t=4751>

Se trata básicamente de un ordenador muy pequeño y de bajo costo que inicialmente estaba enfocado a enseñar a los niños a programar, pero que hasta ahora ésta encontrando su nicho de mercado en los geeks, como yo, a los que nos entretiene trastear con él. Tiene un pequeño procesador ARM a 700 MHz y una GPU capaz de reproducir sin problemas videos en H.264 a 1080p.

¹⁷Tabla 1.6 Características principales de la placa Raspberry PI.

| | |
|------------------------------------|------------------------------------------------------------------|
| Fabricante | Fundación Raspberry PI |
| Tipo | Placa computadora (SBC) |
| Lanzamiento | 29 de Febrero 2012 |
| Sistema Operativo | Linux ARM (Debian, Fedora, Arch Linux), RISC |
| Alimentación | 2.5 W (Modelo A), 3.5 W (Modelo B) |
| CPU | ARM 1176JZF-S (armv6k) A 700mHz |
| CPU | Broadcom VideoCore IV |
| Memoria | 256 MB (Modelo A y primeros en modelo B)/ y 512 MB (Modelo B) |
| Capacidad de Almacenamiento | Tarjeta SD o SDHC |

¹⁷ http://en.wikipedia.org/wiki/Raspberry_Pi

1.5.1.3 Hardware

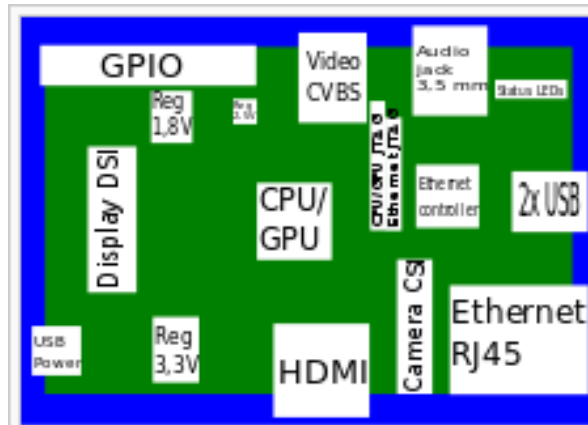


Figura 1. 8 Diagrama que muestra la localización de cada uno de los componentes de la placa Raspberry Pi.

Las ventas iniciales fueron del modelo B. El modelo A solo tiene un puerto USB, carece de controlador Ethernet y cuesta menos que el modelo B, el cual tiene dos puertos USB y controlador Ethernet 10/100.

A pesar que el Modelo A no tiene un puerto RJ45, se puede conectar a una red usando un adaptador USB-Ethernet suministrado por el usuario. Por otro lado, a ambos modelos se puede conectar un adaptador Wi-Fi por USB, para tener acceso a redes inalámbricas o internet. No hay diferencias entre una placa modelo A con un adaptador Ethernet y una de modelo B, ya que ésta última sólo se diferencia de la anterior por llevar el adaptador integrado. Como es típico en los ordenadores modernos, se pueden usar teclados y ratones con conexión USB compatible con Raspberry Pi.

El Raspberry Pi no viene con reloj en tiempo real, por lo que el sistema operativo debe usar un servidor de hora en red, o pedir al usuario la hora en el momento de arrancar el ordenador. Sin embargo se podría añadir un reloj en tiempo real (como el DS1307) con una batería mediante el uso de la interface I²C.

Los esquemas del modelo A y el modelo B fueron lanzados el 20 de abril de 2012 por la fundación.

La aceleración por hardware para la codificación de vídeo (H.264) se hizo disponible el 24 de agosto de 2012, cuando se informó que la licencia permitiría su uso gratuitamente; antes se pensó en anunciarlo cuando se lanzara el módulo de

cámara. También se puso a la venta la capacidad para poder usar la codificación-decodificación de MPEG-2 y Microsoft VC-1. Por otro lado se hizo saber que el ordenador soportaría CEC, permitiendo que pudiera ser controlado mediante un mando a distancia de televisión.

El 5 de septiembre de 2012, se anunció una revisión 2.0 de la placa, que ofrecía un pequeño número de correcciones y mejoras, como unos agujeros de montaje, un circuito para hacer reset, soporte para depuración JTAG, etc.

El 15 de octubre de 2012, la fundación anunció que todos los Raspberry Pi Modelo B serían enviados a partir de ese momento con 512 MB de RAM en vez de 256 MB.⁴

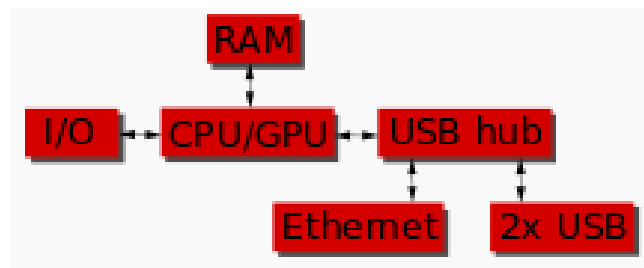


Figura 1.9 Diagrama de bloques del modelo B, en el Modelo A, el puerto USB está conectado directamente al SoC

1.5.1.4 ¹⁸Software

Si bien la fundación Raspberry Pi se ha hecho cargo principalmente de proporcionar el hardware del Modelo B, también ha ayudado para que la comunidad cree distintos sistemas operativos (todos basados en Linux) que satisfagan distintas necesidades.

Ya que el sistema operativo se carga desde una simple tarjeta SD, tener varias distribuciones de Linux listas para ser cambiadas no entraña complicación alguna. De igual manera, es tan sencillo crear una distribución ARM para el Raspberry Pi que cada cierto tiempo se ven nuevas versiones de Linux en los foros del proyecto.

¹⁸ http://es.wikipedia.org/wiki/Raspberry_Pi

Ya que el Modelo B cuenta con dos puertos USB es sencillo conectarle un ratón y un teclado y usarlo como un ordenador cualquiera, aunque sea mucho más lento ordenadores con un precio más elevado. El problema es que si bien se puede usar una interfaz gráfica en el Pi, su limitada memoria y pequeño procesador hace que las cosas funcionen con bastante lentitud. Por eso debe quedar claro que los ordenadores de la fundación Raspberry Pi no fueron creados para remplazar a tu ordenador principal, sino como herramientas de aprendizaje y juego.

1.5.1.5¹⁹ Sistemas operativos

En el momento de escribir este análisis existen tres distribuciones de Linux promocionadas por la fundación Raspberry Pi: Raspbian "wheezy" (basada en Debían), Arch Linux ARM y QtonPi. La primera ha sido clasificada como "para principiantes", y es que se trata de una distribución preconfigurado con administrador de ventanas como LXDE y con algunos navegadores de internet como Midori. La segunda, Arch Linux, tiene como meta proporcionar grandes niveles de control al usuario. Por último, QtonPi es una plataforma embebida para el uso de aplicaciones Qt.

En nuestro caso probamos el funcionamiento de Raspbian, debido a nuestra familiaridad con esa distribución y las optimizaciones hechas para el funcionamiento específico en este dispositivo. El proceso de creación de la tarjeta SD es muy sencillo para una persona con conocimientos básicos de Linux o instalación de sistemas operativos, dado que todos los pasos están enumerados en una wiki. Para poder instalar Raspbian en una tarjeta SD tan sólo hay que saber seguir las instrucciones con cuidado.

La distribución viene con algunas aplicaciones preinstaladas, entre las que se encuentran los navegadores de internet Midori, Dillo y NetSurf y herramientas de programación como IDLE para Python, Scratch y Squeak. Como decíamos, el funcionamiento es lento, hasta el punto de que lanzar cada aplicación requiere varios segundos. A pesar de eso, una vez que la aplicación necesaria está abierta,

¹⁹ http://es.wikipedia.org/wiki/Raspberry_Pi

usarla no representa problema. Eso sí, hay siempre que recordar el limitado espacio de memoria del dispositivo, y por eso la experiencia es mejor si se usa la cantidad mínima posible de aplicaciones a la vez.

La distribución Raspbian cuenta además con algunos juegos desarrollados en Python, como clones de Reversi, Sokoban y Snake. La experiencia de juegos es aceptable, pero se puede sentir la lentitud del ordenador cuando interpreta las teclas seleccionadas.

Además de programación en Python, es posible hacer scripts de Linux en bash con vi., y hasta hemos leído de gente que ha instalado completos servidores web y hace scripts en PHP, si bien nosotros todavía no hemos tenido tiempo de intentarlo.

La configuración de aplicaciones disponibles en los repositorios de Debían funciona tan bien como en cualquier otra instalación de la distribución: tan sólo hay que buscar en apt-get para instalar todo tipo de software en el pequeño ordenador, o por su defecto compilar la fuente directamente con make.

1.5.1.6²⁰ Distribuciones ligeras multipropósito:

- Moebius, distribución ligera ARM HF basada en Debían que usa el repositorio de Raspbian y que cabe en una tarjeta SD de 1GB, usa pocos servicios y está optimizada para usar poca memoria.
- Squeezed Arm Puppy, una versión de Puppy Linux (Puppi) para ARMv6 (sap6) específicamente para Raspberry Pi.¹⁰³

1.5.1.7 Distribuciones ligeras de único propósito:

- Instant WebKiosk, sistema operativo con solo un navegador
- IPFire
- OpenELEC
- Raspbmc
- Xbian

²⁰ http://es.wikipedia.org/wiki/Raspberry_Pi

1.5.1.8 Especificaciones técnicas.

Tabla 1.7 Especificaciones técnicas de la placa Raspberry pi.

| | Modelo A | Modelo B |
|---------------------------------|-------------------------------------------------------------------------------------------|--------------------------------------------------------------------|
| Precio | 25\$ | 35\$ |
| SoC | Broadcom BCM2835 (CPU+GPU+DSP+SDRAM+puerto USB) | |
| CPU | ARM 1176JZF-S a 700MHz (Familia ARM11) | |
| CPU | Broadcom Videocore IV, open GL ES 2.0, -2 y VC-1 (con licencia), 1080p30 H.264/MPGE-4 AVC | |
| Memoria (SDRAM) | 256 MB (compartidos | 512MB (Compartidos con la CPU) desde el 15 de octubre del 2012. |
| Puertos USB 2.0 | 1 | 2 (Vía Hub USB integrado) |
| Salida de video | Conector RCA (PAL y NTSC), HDMI (rev1.3 y 1.4), interfaz DSI para panel LCD. | |
| Salida de Audio | Conector de 3.5mm, HDMI | |
| Almacenamiento Integrado | SD/MMC/ranura para SDIO | |
| Conectividad de red | Ninguna | 10/100 Ethernet (RJ 45) via hub USB. |
| Periféricos de bajo nivel | 8 x GPIO, SPI, I ² C UART | |
| Reloj en tiempo real | Ninguna | |
| Consumo energético | 500mA (2.5W) | 700mA (3.5W) |
| Fuente de alimentación | 5V vía micro USB o GPIO header. | |
| Dimensiones | 85.60mm x 53.98mm (3.370 x 2.125 inch) | |
| Sistemas Operativos soportados. | Debian, Fedora, Arch Linux, Slackware Linux, RISC OS ² | |

1.5.2 ²¹ARDUINO.

Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

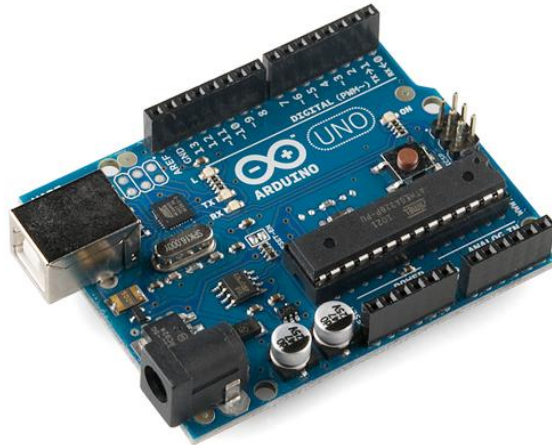
El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida. Los microcontroladores más usados son el Atmega168, Atmega328, Atmega1280, ATmega8 por su sencillez y bajo coste que permiten el desarrollo de múltiples diseños. Por otro lado el software consiste en un entorno de desarrollo que implementa el lenguaje de programación Processing/Wiring y el cargador de arranque (boot loader) que corre en la placa.

Desde octubre de 2012, Arduino se usa también con microcontroladores CortexM3 de ARM de 32 bits⁵, que coexistirán con las más limitadas, pero también económicas AVR de 8 bits. ARM y AVR no son plataformas compatibles a nivel binario, pero se pueden programar con el mismo IDE de Arduino y hacerse programas que compilen sin cambios en las dos plataformas. Eso sí, las microcontroladores CortexM3 usan 3.3V, a diferencia de la mayoría de las placas con AVR que usan mayoritariamente 5V. Sin embargo ya anteriormente se lanzaron placas Arduino con Atmel AVR a 3.3V como la Arduino Fio y existen clónicos de Arduino Nano y Pro como Meduino en que se puede conmutar el voltaje.

Arduino se puede utilizar para desarrollar objetos interactivos autónomos o puede ser conectado a software del ordenador (por ejemplo: Macromedia Flash, Processing, Max/MSP, Pure Data). Las placas se pueden montar a mano o adquirirse. El entorno de desarrollo integrado libre se puede descargar gratuitamente.

Al ser open-hardware, tanto su diseño como su distribución es libre. Es decir, puede utilizarse libremente para el desarrollo de cualquier tipo de proyecto sin haber adquirido ninguna licencia.

²¹ <http://arduino.cc/en/Main/arduinoBoardUno>



²²Figura 1.10 Placa Arduino Uno.

1.5.2.1 Descripción.

Arduino Uno es la versión mejorada de su predecesor Duemilanove. Incluye función de autoreset, protección de sobrecargas, conector USB para programarlo, totalmente montado con componentes miniatura SMD (salvo el microcontrolador, para poder cambiarlo fácilmente) y nuevo bootloader OptiBoot a 155kbps.

El Arduino Uno es 100% compatible con Arduino Duemilanove pero incorpora una autoselección del voltaje de alimentación (DC/USB) gracias a un chip MOSFET incluido en la placa. Además, dispone del nuevo bootloader OptiBoot que permite cargar programas a 115Kbps. El bootloader también ha sido reducido en tamaño ya que tan sólo ocupa 512bytes, por lo que tendremos aún más espacio para programar.

Se entrega con el nuevo chip Atmega328 de AVR con 32 KB de memoria de programa en lugar de FTDI chip con 16 KB, RAM de 2KB (antes 1KB) y EEPROM de 1 KB (antes 512 bytes). Eso permite la carga más rápida de programas, no requiere "drivers" para MAC o Linux (el fichero de info para Windows es necesario), y además permite la posibilidad que el Arduino Uno aparezca como el ratón, keyboard, joystick, etc.

²² <http://www.bricogeek.com/shop/arduino/305-arduino-uno.html>

La placa se entrega completamente ensamblada y probada con un microcontrolador AVR ATmega328 con un cristal de cuarzo de 16Mhz. El microcontrolador se entrega con un bootloader que permite su programación sin necesidad de ningún tipo de programado externo.

Cuando se habla de Arduino, es necesario tener en cuenta que este se halla constituido por tres elementos fundamentales y que todos ellos se denominan bajo el mismo nombre (Arduino), estos componentes son:

- El hardware o placas de Arduino
- El entorno de programación o IDE
- El lenguaje de programación.

En general estos componentes fueron creados con la finalidad de tener una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles y fáciles de usar. Se creó para artistas, diseñadores, aficionados y cualquiera interesado en crear entornos u objetos interactivos.

1.5.2.2²³ El hardware

El hardware, en su mayoría, está constituido por una baquelita electrónica desarrollada en torno a un microcontrolador de Atmel, estas placas se pueden adquirir ensambladas o construirlas directamente ya que en la página se encuentran los planos electrónicos y la licencia del producto lo permite, sin embargo resultan valer aproximadamente lo mismo y nunca quedan con la calidad de las placas originales. En la actualidad a Colombia están ingresando muchas placas de Arduino de origen chino, las cuales bajan un poco la calidad de los materiales pero mantienen las mismas funciones que las placas originales provenientes de Italia, esta es la razón por la cual algunos vendedores en mercado libre y tiendas online las ofrecen a un precio bastante bajo.

Estas placas han ido evolucionando así como el software y es mejor comprar de las más recientes como el Leonardo, la Arduino Uno R3 y las nanos basadas en el ATmega328, pero si lo que se necesitan son entradas y salidas en gran cantidad, lo

²³ <http://arduino.cc/en/Main/arduinoBoardUno>

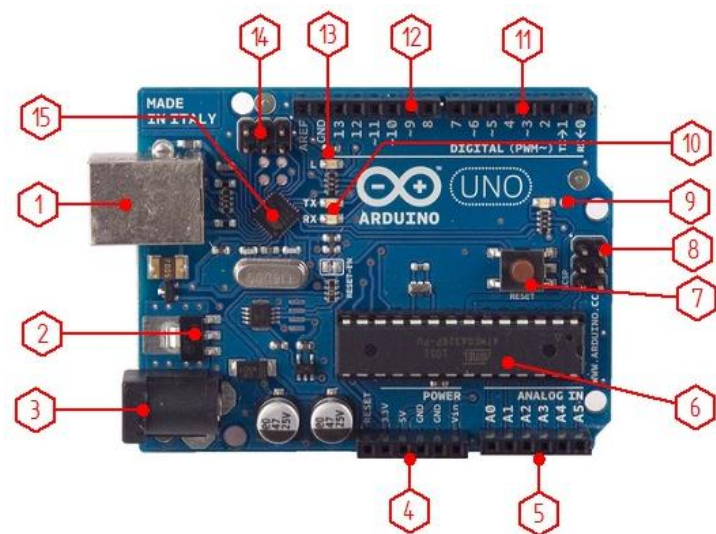
mejor es la Arduino Mega. Las primeras placas utilizaban un chip ft232 para comunicarse por puerto USB al computador y un micro para ser programado, luego se utilizó un microcontrolador especial para cumplir esta función como en el caso del Arduino uno que tienen un micro para la comunicación y otro para ser programado, actualmente se usa un único microcontrolador que se encarga de llevar a cabo la comunicación y sobre el que también se descargan las instrucciones a ejecutar, este es el caso del Arduino Leonardo por ejemplo.

Los lectores deben entender que en principio es posible crear una placa de desarrollo alrededor de cualquier microcontrolador y lo que se logra con las placas de Arduino es tener el microcontrolador listo para conectar a los sensores y actuadores que requerimos y programarlo, esto le confiere una gran velocidad en el desarrollo de proyectos ya que cinco minutos luego de conectar tu placa por primera vez al computador ya puedes estar descargándole el primer programa aun sin saber programar. Por otro lado se tienen una serie de aditamentos llamados Shell (escudos o cáscaras) los cuales le dan más funcionalidad a las placas.

Al escoger un Arduino, las principales características a tener en cuenta en cuanto al hardware son:

- Tamaño, por ejemplo para diseño de dispositivos de uso cotidiano en ocasiones es mucho mejor utilizar los arduinos mini o nano que son de poco tamaño.
- Número de entradas/ salidas, es necesario saber cuántas entradas y cuantas salidas serán necesarias para el proyecto
- Capacidad de memoria del microcontrolador, por si realizas programas muy ... muy grandes
- Voltaje de operación.

1.5.2.3 Principales partes que forman una Placa Arduino Uno.



²⁴Figura 1. 11 Partes constitutivas de la placa Arduino.

El hardware del Arduino uno R2, se muestran en la figura de arriba aunque no necesariamente todos los arduinos son iguales, a continuación describo a groso modo cada una de estas.

1. Conector USB, que puede ser tipo B o mini, este provee la comunicación para la programación y la toma de datos, también provee una fuente de 5VDC para alimentar al Arduino, pero de baja corriente por lo que no sirve para alimentar motores grandes por ejemplo. Siempre que adquieran una placa de Arduino no olviden pedir el cable de conexión USB pues este representa un \$7.00 adicional.
2. Regulador de voltaje de 5V, se encarga de convertir el voltaje ingresado por el plug 3, en un voltaje de 5V regulado. necesario para el funcionamiento de la placa y para alimentar circuitos externos.
3. Plug de conexión para fuente de alimentación externa, el voltaje que se suministra por aquí debe ser directo y estar entre 6V y 18V, incluso 20V, generalmente se usa un adaptador, pero debe tener cuidado de que el

²⁴ <http://www.cortoc.com/p/arduino.html>

terminal del centro del plug quede conectado a positivo ya que algunos adaptadores traen la opción de intercambiar la polaridad de los cables.

4. Puerto de conexiones; constituido por 6 pines de conexión con las siguientes funciones: RESET, permite resetear el microcontrolador al enviarle un cero lógico. Pin 3.3V, este pin provee una fuente de 3.3VDC para conectar dispositivos externos como en la protoboard por ejemplo. Pin 5V, es una fuente de 5VDC para conectar dispositivos externos. Dos pines GND, que proveen la salida de cero voltios para dispositivos externos. Pin Vin, este pin está conectado con el positivo del plug 3 por lo que se usa para conectar la alimentación de la placa con una fuente externa de entre 6 y 12VDC en lugar del plug 3 o la alimentación por el puerto USB. Este puerto esta modificado en la versión R3 de Arduino Uno.
5. Puerto de entradas análogas, aquí se conectan las salidas de los sensores análogos. Estos pines solo funcionan como entradas recibiendo voltajes entre cero y cinco voltios directos.
6. Microcontrolador Atmega 328, es el microcontrolador implementado en los Arduino uno y sobre el cual vamos a programar, en la versión SMD del Arduino uno R2, se usa el mismo microcontrolador pero en montaje superficial, en este caso las únicas ventajas que se me ocurren son la reducción del peso y ganar un poco de espacio.
7. Botón de RESET, este botón así como el pin mencionado anteriormente permiten resetear el microcontrolador haciendo que reinicie el programa. En la versión R3 este pulsador se ubica arriba del conector USB, esto es un acierto pues al colocarle las Shield encima del Arduino, se perdía la opción de resetear dado que este pulsador quedaba tapado.
8. Pines de programación ICSP, son usados para programar microcontroladores en protoboard o sobre circuitos impresos sin tener que retirarlos de su sitio.
9. LED ON, enciende cuando el Arduino está encendido.
10. LEDs de recepción y transmisión, estos se encienden cuando la tarjeta se comunica con el PC. El Tx indica transmisión de datos y el Rx recepción.

11. Puerto de conexiones, está constituido por los pines de entradas o salidas digitales desde la cero hasta la 7. La configuración como entrada o salida debe ser incluida en el programa. Cuando se usa la terminal serial es conveniente no utilizar los pines cero (Rx) y uno (Tx). Los pines 3, 5 y 6 están precedidos por el símbolo, lo que indica que permiten su uso como salidas controladas por ancho de pulso PWM.
12. Puerto de conexiones, incluye 5 entradas o salidas adicionales (de la 8 a la 12), las salidas 9, 10 y 11 permiten control por ancho de pulso; la salida 13 es un poco diferente pues tiene conectada una resistencia en serie, lo que permite conectar un led directamente entre ella y tierra. Finalmente hay una salida a tierra GND y un pin AREF que permite ser empleado como referencia para las entradas análogas.
13. Este led indica el estado del pin 13.
14. No sé exactamente la función de estos pines.
15. Chip de comunicación que permite la conversión de serial a USB.

Arduino puede tomar información del entorno a través de sus pines de entrada de toda una gama de sensores y puede afectar aquello que le rodea controlando luces, motores y otros actuadores. El microcontrolador en la placa Arduino (de la marca atmel) se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). Una vez descargados en las placas Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a un ordenador, si bien tienen la posibilidad de hacerlo y comunicar con diferentes tipos de software (por ejemplo Flash, Processing, MaxMSP).

Las placas pueden ser hechas a mano (de lo que se consigue abundante material en la red) o compradas montadas de fábrica; el software puede ser descargado de forma gratuita de las páginas oficiales de Arduino, sin embargo recomiendo descargar de la página en inglés en donde obtendrán la versión más actualizada. Los ficheros de diseño de referencia (CAD) están disponibles bajo una licencia abierta, así pues eres libre de adaptarlos a tus necesidades.

Arduino recibió una Mención Honorífica en la sección Digital Communities de la edición del 2006 del Ars Electrónica Prix. El equipo Arduino (Arduino team) es: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis. Credits.

1.5.2.4 ²⁵Ventajas y desventajas del Arduino.

Ventajas de Arduino.

- Dado que tanto el software como el hardware es libre, se hacen ideal para los proyectos de la universidad y de pequeñas empresas sin temor a estar violando los derechos de autor de alguien.
- Las placas Arduino son más asequibles comparadas con otras plataformas de microcontroladores.
- Multi-Plataforma - El software de Arduino funciona en los sistemas operativos Windows, Macintosh OSX y Linux.
- Entorno de programación simple y directo - El entorno de programación de Arduino es fácil de usar para principiantes y lo suficientemente flexible para los usuarios avanzados.
- Software ampliable y de código abierto- El software Arduino está publicado bajo una licencia libre y preparada para ser ampliado por programadores experimentados. El lenguaje puede ampliarse a través de librerías de C++, y si se está interesado en profundizar en los detalles técnicos, se puede dar el salto a la programación en el lenguaje AVR C en el que está basado.
- Hardware ampliable y de Código abierto - Arduino está basado en los microcontroladores ATMEGA168, ATMEGA328 y ATMEGA1280. Los planos de los módulos están publicados bajo licencia Creative Commons, por lo que diseñadores de circuitos con experiencia pueden hacer su propia versión del módulo, ampliándolo u optimizándolo. Incluso usuarios

²⁵ <https://sites.google.com/site/temasdedisenoymanufactura/arduino>

relativamente inexpertos pueden construir la versión para placa de desarrollo para entender cómo funciona y ahorrar algo de dinero.

1.5.2.5 Características Principales de la placa Arduino Uno:

- Microcontrolador: ATmega328 microcontroller
- Voltaje: 5V
- Voltaje de entrada (recomendado): 7-12V
- Voltaje de entrada (limite): 6-20V
- Pines Digitales de entrada y salida: 14 (6 de ellos con salida PWM)
- Pines de entrada Analógicos: 6
- Memoria: 32 KB (2 KB utilizados por bootloader)
- Corriente DC por Pin: 40 mA
- Corriente DC por Pin de 3.3V: 50 mA
- SRAM: 2 KB
- EEPROM: 1 KB
- Velocidad de reloj: 16 MHz

1.6 ²⁶COMUNICACIÓN DEL PC CON ELEMENTOS EXTERNOS.

En la informática, un puerto es una forma genérica de denominar a una interfaz a través de la cual los diferentes tipos de datos se pueden enviar y recibir. Dicha interfaz puede ser de tipo físico, o puede ser a nivel de software (por ejemplo, los puertos que permiten la transmisión de datos entre diferentes ordenadores.

1.6.1 Puerto Serie.

El puerto en serie de un ordenador es un adaptador asíncrono utilizado para poder intercomunicar varios ordenadores entre sí. Un puerto serie recibe y

²⁶ <http://es.scribd.com/doc/35568727/Puertos-y-Conectores-de-La-PC>

envía información fuera del ordenador mediante un determinado software de comunicación o un drive del puerto serie.

Tabla 1.8 Pines de una comunicación en serie.

| Pin | Pin | Nombre | Descripción |
|-----|-----|--------|------------------------------|
| 1 | 8 | CD | Detección de portadora. |
| 2 | 3 | RXD | Recepción de datos |
| 3 | 2 | TXD | Transmisión de datos. |
| 4 | 20 | DTR | Terminal de datos preparado. |
| 5 | 7 | GND | Masa del sistema. |
| 6 | 6 | DSR | Puesta de datos preparado. |
| 7 | 4 | RTS | Solicitud de envío. |
| 8 | 5 | CTS | Borrado para enviar. |
| 9 | 22 | RI | Indicador de llamada. |

1.6.2 Puerto Paralelo.

La transmisión de datos paralela consiste en enviar datos en forma simultánea por varios canales (hilos). Los puertos paralelos en los PC pueden utilizarse para enviar 8 bits (un octeto) simultáneamente por 8 hilos.



Figura 1.12 Conexión del puerto paralelo.

1.6.3 Puerto USB

El Universal Serial Bus (USB) (bus universal en serie BUS) es un estándar industrial desarrollado en los años 1990 que define los cables, conectores y protocolos usados en un bus para conectar, comunicar y proveer de alimentación eléctrica entre ordenadores y periféricos y dispositivos electrónicos.

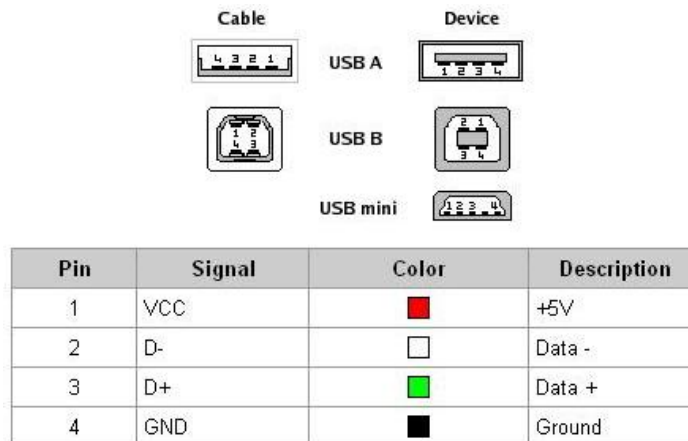


Figura 1.13 Tipos y polaridad de los puertos USB

1.6.3.1 Características del puerto USB

La versión USB 1.0 Aparece en el mercado, junto con el lanzamiento del microprocesador Intel® Pentium II en 1997.

Cada puerto, permite conectar hasta 127 dispositivos externos, pero solo se recomiendan como máximo 8, porque se satura la línea del puerto y se ralentiza el sistema al tener que administrarse todos simultáneamente.

Cuenta con tecnología "Plug&Play" la cual permite conectar, desconectar y reconocer dispositivos sin necesidad de reiniciar o apagar la computadora.

Las versiones USB 1.X y USB 2.0 transmiten en un medio unidireccional los datos, esto es solamente se envía o recibe datos en un sentido a la vez, mientras que la versión USB 3 cuenta con un medio Duplex que permite enviar y recibir datos de manera simultánea.

A pesar de que el puerto USB 3, está actualmente integrado ya en algunas placas de nueva generación, aún no hay dispositivos comerciales/populares para esta tecnología.

1.7 DISEÑO Y PUBLICACION DE LA PÁGINA WEB.

1.7.1 El Internet.

Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, Estados Unidos.

Uno de los servicios que más éxito ha tenido en Internet ha sido la World Wide Web(WWW, o "la Web"), hasta tal punto que es habitual la confusión entre ambos términos. La WWW es un conjunto de protocolos que permite, de forma sencilla, la consulta remota de archivos de hipertexto. Ésta fue un desarrollo posterior (1990) y utiliza Internet como medio de transmisión.

Existen, por tanto, muchos otros servicios y protocolos en Internet, aparte de la Web: el envío de correo electrónico (SMTP), la transmisión de archivos (FTP y P2P), las conversaciones en línea (IRC), la mensajería instantánea y presencia, la transmisión de contenido y comunicación multimedia - telefonía (VoIP), televisión (IPTV)-, los boletines electrónicos (NNTP), el acceso remoto a otros dispositivos (SSH y Telnet) o los juegos en línea.



Figura 1.14 Enlace de datos a través del Internet.

1.7.2²⁷ Como conectarse a Internet

Acceso a Internet o Conexión a Internet es el mecanismo de enlace con que una computadora o red de computadoras cuenta para conectarse a Internet, lo que les permite visualizar las páginas web desde un navegador y acceder a otros servicios que ofrece esta red. Se puede acceder a internet desde una Conexión por línea conmutada, Banda ancha fija (a través de cable coaxial, cables de fibra óptica o cobre), WiFi, vía satélite, Banda Ancha Móvil y teléfonos celulares con tecnología 3G/4G. Las empresas que otorgan acceso a Internet reciben el nombre de proveedores (ISP).

1.7.3²⁸ Servicios web.

Un servicio web (en inglés, Web Services) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios Web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles para definir de manera más exhaustiva estos estándares. Es una máquina que atiende las peticiones de los clientes web y les envía los recursos solicitados.

²⁷ http://es.wikipedia.org/wiki/Conexi%C3%B3n_a_Internet

²⁸ http://es.wikipedia.org/wiki/Servicio_web



Figura 1.15 Conexión cliente servidor por medio de Internet.

1.7.4 Transmisión de datos por internet.

Una red de ordenadores permite conectar a los ordenadores que la forman con la finalidad de compartir información, como documentos o bases de datos, o recursos físicos, como impresoras o unidades de disco. Las redes suelen clasificarse según su extensión en:

- LAN (Local Área Network): Son las redes de área local. La extensión de este tipo de redes suele estar restringida a una sala edificio, aunque también podría utilizarse para conectar dos más edificios próximos.
- WAN (Wide Área Network): Son redes que cubren un espacio muy amplio, conectando a ordenadores de una ciudad o un país completo. Para ello se utilizan las líneas de teléfono y otros medios de transmisión más sofisticados, como pueden ser las microondas. La velocidad de transmisión suele ser inferior que en las redes locales.

Lo que se conoce como Internet es en realidad una red de redes, la interconexión de otras redes independientes de manera que puedan compartir información entre ellas a lo largo de todo el planeta. Para ello es necesario el uso de un protocolo de comunicaciones común. El protocolo que proporciona la compatibilidad necesaria para la comunicación en Internet es el TCP/IP.

1.7.4.1 ²⁹Modelo OSI

El modelo OSI (Open Systems Interconnection) es utilizado por prácticamente la totalidad de las redes del mundo. Este modelo fue creado por el ISO (Organización Internacional de Normalización), y consiste en siete niveles o capas donde cada una de ellas define las funciones que deben proporcionar los protocolos con el propósito de intercambiar información entre varios sistemas. Esta clasificación permite que cada protocolo se desarrolle con una finalidad determinada, lo cual simplifica el proceso de desarrollo e implementación. Cada nivel depende de los que están por debajo de él, y a su vez proporciona alguna funcionalidad a los niveles superiores.

³⁰Tabla 1.9 Niveles del modelo OSI

| | |
|--------------|----------------------------------------------------------------------------------------------------------|
| Aplicación | El nivel de aplicación es el destino final de los datos donde se proporcionan los servicios al usuario. |
| Presentación | Se convierten e interpretan los datos que se utilizarán en el nivel de aplicación. |
| Sesión | Encargado de ciertos aspectos de la comunicación como el control de los tiempos. |
| Transporte | Transporta la información de una manera fiable para que llegue correctamente a su destino. |
| Red | Nivel encargado de encaminar los datos hacia su destino eligiendo la ruta más efectiva. |
| Enlace | Enlace de datos. Controla el flujo de los mismos, la sincronización y los errores que puedan producirse. |
| Físico | Se encarga de los aspectos físicos de la conexión, tales como el medio de transmisión o el hardware. |

²⁹ <http://www.monografias.com/trabajos13/modosi/modosi.shtml>

³⁰ http://es.wikipedia.org/wiki/Modelo_OSI

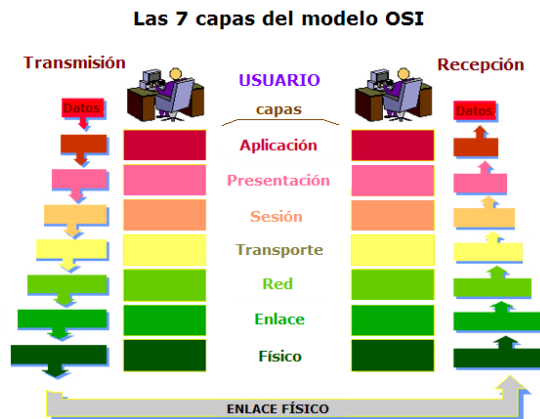


Figura 1.16 Enlace del modelo OSI

1.7.4.2 ³¹Arquitectura cliente servidor

La arquitectura cliente-servidor es una forma específica de diseño de aplicaciones, aunque también se conoce con este nombre a los ordenadores en los que se estas aplicaciones son ejecutadas. Por un lado, el cliente es el ordenador que se encarga de efectuar una petición o solicitar un servicio.

El cliente no posee control sobre los recursos, sino que es el servidor el encargado de manejarlos. Por otro lado, el ordenador remoto que actúa como servidor evalúa la petición del cliente y decide aceptarla o rechazarla consecuentemente.

Una vez que el servidor acepta el pedido la información requerida es suministrada al cliente que efectuó la petición, siendo este último el responsable de proporcionar los datos al usuario con el formato adecuado.

Finalmente debemos precisar que cliente y servidor no tienen que estar necesariamente en ordenadores separados, sino que pueden ser programas diferentes que se ejecuten en el mismo ordenador.

³¹ <http://www.monografias.com/trabajos5/datint/datint.shtml>



³²Figura 1.17 Arquitectura cliente servidor.

1.7.4.3 Protocolo TCP/IP

TCP/IP es el protocolo común utilizado por todos los ordenadores conectados a Internet, de manera que éstos puedan comunicarse entre S. Hay que tener en cuenta que en Internet se encuentran conectados ordenadores de clases muy diferentes y con hardware y software incompatibles en muchos casos, además de todos los medios y formas posibles de conexión. Aquí se encuentra una de las grandes ventajas del TCP/IP, pues este protocolo se encargará de que la comunicación entre todos sea posible. TCP/IP es compatible con cualquier sistema operativo y con cualquier tipo de hardware.

TCP/IP no es un único protocolo, sino que es en realidad lo que se conoce con este nombre es un conjunto de protocolos que cubren los distintos niveles del modelo OSI. Los dos protocolos más importantes son el TCP (Transmission Control Protocol) y el IP (Internet Protocol), que son los que dan nombre al conjunto. En Internet se diferencian cuatro niveles o capas en las que se agrupan los protocolos, y que se relacionan con los niveles OSI de la siguiente manera:

- **Aplicación:** Se corresponde con los niveles OSI de aplicación, presentación y sesión. Aquí se incluyen protocolos destinados a proporcionar servicios, tales como correo electrónico (SMTP), transferencia de ficheros (FTP), conexión remota (TELNET) y otros más recientes como el protocolo HTTP (Hypertext Transfer Protocol).

³² <http://oscarbasededatos2012.blogspot.com/2012/11/arquitectura-clienteservidor.html>

- Transporte: Coincide con el nivel de transporte del modelo OSI. Los protocolos de este nivel, tales como TCP y UDP, se encargan de manejar los datos y proporcionar la fiabilidad necesaria en el transporte de los mismos.
- Internet: Es el nivel de red del modelo OSI. Incluye al protocolo IP, que se encarga de enviar los paquetes de información a sus destinos correspondientes. Es utilizado con esta finalidad por los protocolos del nivel de transporte.
- Enlace: Los niveles OSI correspondientes son el de enlace y el nivel físico. Los protocolos que pertenecen a este nivel son los encargados de la transmisión a través del medio físico al que se encuentra conectado cada host, como puede ser una línea punto a punto o una red Ethernet.

El TCP/IP necesita funcionar sobre algún tipo de red o de medio físico que proporcione sus propios protocolos para el nivel de enlace de Internet. Por este motivo hay que tener en cuenta que los protocolos utilizados en este nivel pueden ser muy diversos y no forman parte del conjunto TCP/IP. Sin embargo, esto no debe ser problemático puesto que una de las funciones y ventajas principales del TCP/IP es proporcionar una abstracción del medio de forma que sea posible el intercambio de información entre medios diferentes y tecnologías que inicialmente son incompatibles.

Para transmitir información a través de TCP/IP, ésta debe ser dividida en unidades de menor tamaño. Esto proporciona grandes ventajas en el manejo de los datos que se transfieren y, por otro lado, esto es algo común en cualquier protocolo de comunicaciones. En TCP/IP cada una de estas unidades de información recibe el nombre de "datagrama" (datagram), y son conjuntos de datos que se envían como mensajes independientes.

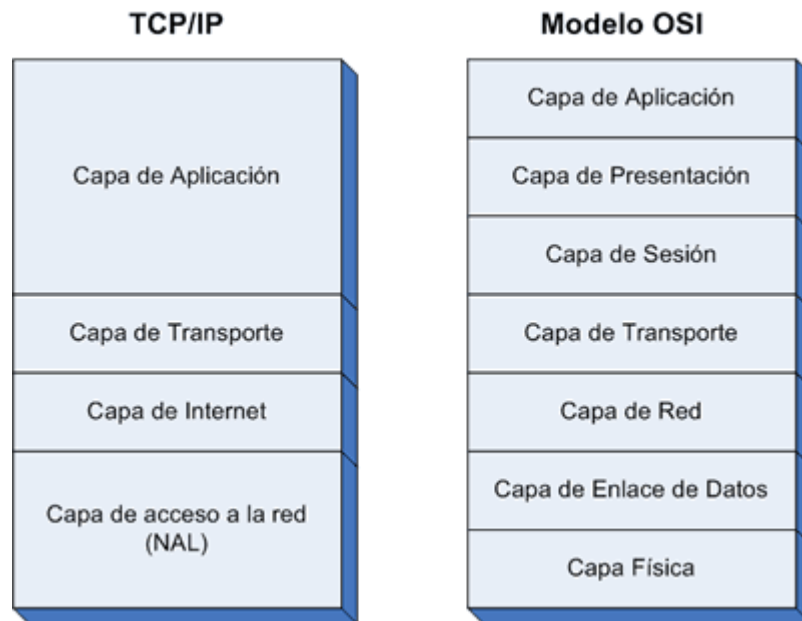


Figura 1.18 Comparación de capas del modelo OSI y TCP/IP

1.7.4.4 ³³IP (Internet Protocolo)

El IP es un protocolo que pertenece al nivel de red, por lo tanto, es utilizado por los protocolos del nivel de transporte como TCP para encaminar los datos hacia su destino. IP tiene únicamente la misión de encaminar el datagrama, sin comprobar la integridad de la información que contiene. Para ello se utiliza una nueva cabecera que se antepone al datagrama que se está tratando. Suponiendo que el protocolo TCP ha sido el encargado de manejar el datagrama antes de pasarlo al IP, la estructura del mensaje una vez tratado quedaría así:

| | | |
|--------------------------|---------------------------|-------|
| Cabecera IP (20 byte) | Cabecera TCP (20 byte) | Datos |
|--------------------------|---------------------------|-------|

La cabecera IP tiene un tamaño de 160 bit y está formada por varios campos de distinto significado. Estos campos son:

Versión: Número de versión del protocolo IP utilizado. Tendrá que tener el valor 4. Tamaño: 4 bit.

³³ <http://www.monografias.com/trabajos32/transmision-informacion-internet/transmision-informacion-internet.shtml>

Longitud de la cabecera: (Internet Header Length, IHL) Especifica la longitud de la cabecera expresada en el número de grupos de 32 bit que contiene. Tamaño: 4 bit.

Tipo de servicio: El tipo o calidad de servicio se utiliza para indicar la prioridad o importancia de los datos que se envían, lo que condicionará la forma en que éstos serán tratados durante la transmisión. Tamaño: 8 bit.

Longitud total: Es la longitud en bytes del datagrama completo, incluyendo la cabecera y los datos. Como este campo utiliza 16 bit, el tamaño máximo del datagrama no podrá superar los 65.535 bytes, aunque en la práctica este valor será mucho más pequeño. Tamaño: 16 bit.

Identificación: Valor de identificación que se utiliza para facilitar el ensamblaje de los fragmentos del datagrama. Tamaño: 16 bit.

Flags: Indicadores utilizados en la fragmentación. Tamaño: 3 bit.

Fragmentación: Contiene un valor (offset) para poder ensamblar los datagramas que se hayan fragmentado. Está expresado en número de grupos de 8 bytes (64 bit), comenzando con el valor cero para el primer fragmento. Tamaño: 16 bit.

Límite de existencia: Contiene un número que disminuye cada vez que el paquete pasa por un sistema. Si este número llega a cero, el paquete será descartado. Esto es necesario por razones de seguridad para evitar un bucle infinito, ya que aunque es bastante improbable que esto suceda en ad. Tamaño: 8 bit.

Protocolo: El número utilizado en este campo sirve para indicar a qué protocolo pertenece el datagrama que se encuentra a continuación de la cabecera IP, de manera que pueda ser tratado correctamente cuando llegue a su destino. Tamaño: 8 bit.

Comprobación: El campo de comprobación (checksum) es necesario para verificar que los datos contenidos en la cabecera IP son correctos. Por razones de eficiencia este campo no puede utilizarse para comprobar los datos incluidos a continuación, sino que estos datos de usuario se comprobarán posteriormente a partir del campo de comprobación de la cabecera siguiente, y que corresponde al nivel de transporte. Este campo debe calcularse de nuevo cuando cambia alguna opción de la cabecera, como puede ser el límite de existencia. Tamaño: 16 bit.

Dirección de origen: Contiene la dirección del host que envía el paquete. Tamaño: 32 bit.

Dirección de destino: Esta dirección es la del host que recibirá la información. Los routers o gateways intermedios deben conocerla para dirigir correctamente el paquete. Tamaño: 32 bit.

1.7.4.5 ³⁴Direcciones Internet (IP)

El protocolo IP identifica a cada ordenador que se encuentre conectado a la red mediante su correspondiente dirección. Esta dirección es un número de 32 bit que debe ser único para cada host, y normalmente suele representarse como cuatro cifras de 8 bit separadas por puntos.

La dirección de Internet (IP Address) se utiliza para identificar tanto al ordenador en concreto como la red a la que pertenece, de manera que sea posible distinguir a los ordenadores que se encuentran conectados a una misma red. Con este propósito, y teniendo en cuenta que en Internet se encuentran conectadas redes de tamaños muy diversos, se establecieron tres clases diferentes de direcciones, las cuales se representan mediante tres rangos de valores:

- Clase A: Son las que en su primer byte tienen un valor comprendido entre 1 y 126, incluyendo ambos valores. Estas direcciones utilizan únicamente este primer byte para identificar la red, quedando los otros tres bytes disponibles para cada uno de los hosts que pertenezcan a esta misma red. Esto significa que podrán existir más de dieciséis millones de ordenadores en cada una de las redes de esta clase. Este tipo de direcciones es usado por redes muy extensas, pero hay que tener en cuenta que sólo puede haber 126 redes de este tamaño. ARPA net es una de ellas, existiendo además algunas grandes redes comerciales, aunque son pocas las organizaciones que obtienen una dirección de "clase A". Lo normal para las grandes organizaciones es que utilicen una o varias redes de "clase B".
- Clase B: Estas direcciones utilizan en su primer byte un valor comprendido entre 128 y 191, incluyendo ambos. En este caso el identificador de la red se obtiene de los dos primeros bytes de la dirección, teniendo que ser un valor entre

³⁴ http://es.wikipedia.org/wiki/Direcci%C3%B3n_IP

128.1 y 191.254 (no es posible utilizar los valores 0 y 255 por tener un significado especial). Los dos últimos bytes de la dirección constituyen el identificador del host permitiendo, por consiguiente, un número máximo de 64516 ordenadores en la misma red. Este tipo de direcciones tendría que ser suficiente para la gran mayoría de las organizaciones grandes. En caso de que el número de ordenadores que se necesita conectar fuese mayor, sería posible obtener más de una dirección de "clase B", evitando de esta forma el uso de una de "clase A".

- Clase C: En este caso el valor del primer byte tendrá que estar comprendido entre 192 y 223, incluyendo ambos valores. Este tercer tipo de direcciones utiliza los tres primeros bytes para el número de la red, con un rango desde 192.1.1 hasta 223.254.254. De esta manera queda libre un byte para el host, lo que permite que se conecten un máximo de 254 ordenadores en cada red. Estas direcciones permiten un menor número de host que las anteriores, aunque son las más numerosas pudiendo existir un gran número redes de este tipo (más de dos millones).

Tabla 1.10 Direcciones IP de Internet.

| Clase | Primer Byte | Identificación de red | Identificación de Hosts | Numero de redes | Numero de hosts |
|-------|-------------|-----------------------|-------------------------|-----------------|-----------------|
| A | 1...126 | 1 byte | 3 byte | 126 | 16.387.064 |
| B | 128...191 | 2 byte | 2 byte | 16.256 | 64.516 |
| C | 192...223 | 3 byte | 1 byte | 2.064.512 | 254 |

En la clasificación de direcciones anterior se puede notar que ciertos números no se usan. Algunos de ellos se encuentran reservados para un posible uso futuro, como es el caso de las direcciones cuyo primer byte sea superior a 223 (clases D y E, que aún no están definidas), mientras que el valor 127 en el primer byte se utiliza en algunos sistemas para propósitos especiales. También es importante notar que los valores 0 y 255 en cualquier byte de la dirección no pueden usarse normalmente por tener otros propósitos específicos. El número 0 está reservado para las máquinas que no conocen su dirección, pudiendo utilizarse tanto en la identificación de red para máquinas que aún no conocen el número de red a la que

se encuentran conectadas, en la identificación de host para máquinas que aún no conocen su número de host dentro de la red, o en ambos casos.

El número 255 tiene también un significado especial, puesto que se reserva para el broadcast. El broadcast es necesario cuando se pretende hacer que un mensaje sea visible para todos los sistemas conectados a la misma red. Esto puede ser útil si se necesita enviar el mismo datagrama a un número determinado de sistemas, resultando más eficiente que enviar la misma información solicitada de manera individual a cada uno. Otra situación para el uso de broadcast es cuando se quiere convertir el nombre por dominio de un ordenador a su correspondiente número IP y no se conoce la dirección del servidor de nombres de dominio más cercano.

³⁵Tabla 1.11 Representación decimal de direcciones de Internet

| | Octeto 1 | Octeto 2 | Octeto 3 | | Rango de direcciones |
|--------------------------------|---------------------|------------------------|----------------------|--------------------|--------------------------------|
| Clase A | ID red 1 a 127 | ID host 0 a 255 | ID host 0 a 255 | ID host 0 a 255 | 1.0.0.0 a 127.255.255.255 |
| Clase B | ID red 128 a 191 | ID red 0 a 255 | ID host 0 a 255 | ID host 0 a 255 | 128.0.0.0 a 191.255.255.255 |
| Clase C | ID red 192 a 223 | ID red 0 a 255 | ID red 0 a 255 | ID host 1 a 254 | 192.0.0.0 a 223.255.255.255 |
| Clase D (Multicast) | 224 a 239 | Direcciones 0 a 255 | Multicast 0 a 255 | 1 a 254 | 224.0.0.0 a 239.255.255.255 |
| Clase E (Reservado) | 240 a 255 | 0 a 255 | 0 a 255 | 1 a 254 | 240.0.0.0 a 255.255.255.255 |

³⁵ <http://www.monografias.com/trabajos29/direccionamiento-ip/direccionamiento-ip.shtml>

1.7.5³⁶ Medios de acceso a internet

Cuando se inventó el internet, como un proyecto militar del ejército de Estados Unidos, se usaban medios civiles, como las líneas telefónicas, para el flujo de la información. Hoy día, los medios son otros, más modernos, más rápidos y, en algunos casos, más accesibles.

Enumeramos a continuación los medios de que se dispone, para acceder a la internet en el presente, separándolos en dos categorías, doméstico y comercial, no porque la tecnología sea el motivo de la separación de ambas, sino la diferencia en el costo monetario entre unas y otras.

1.7.5.1 Para clientes domésticos:

- **Línea telefónica-** El acceso a internet por medio de las líneas de teléfonos, es la manera más antigua de contactarse. La evolución de ese medio, fue en incrementos de la velocidad de acceso, hasta el límite de 56 K (56,000 bauds por segundo). Esa limitación no es física, ni tecnológica, sino una regulación de la Comisión Federal de Comunicaciones (FCC, por sus siglas en inglés).
- **ADSL-** Este medio es el cumplimiento parcial de una promesa de los años setenta, en que los servicios de telefonía, televisión y radio, se iban a servir por medio de una única línea de fibra óptica. Eso se conoció como Red Digital de Servicios integrados (ISDN, por sus siglas en inglés). Hoy, el DSL es el medio de acceso por medio de las líneas telefónicas, más rápido (y más costoso) que el dial-up.
- **Cable Digital-** Este medio se logra a través de la misma línea que sirve la televisión por cable. A pesar que puede lograr unas velocidades de conexión muchos más rápidas que el dial-up, es comparable al ADSL.
- **Red Celular-** Para los consumidores domésticos, esta es la forma de acceso que se está popularizando rápidamente, por la conveniencia de la movilidad. Se accede al internet mediante la misma red de telefonía móvil. Originalmente, se logra por medio de un teléfono móvil, pero también se ofrece un dispositivo dedicado al

³⁶ <http://tecnosaurios.blogspot.com/2011/10/medios-de-acceso-internet.html>

único propósito de conectarse al internet. Su precio es más elevado que el ADSL, pero las velocidades de acceso son más altas.

- ³⁷**Modem Banda Ancha-** también conocido como **BAM**, es un servicio de Internet Móvil que ofrece multitud de Proveedores de Telefonía móvil como claro, Movistar, Alegro y CNT en Ecuador.

Esta tecnología permite obtener internet en cualquier lugar y momento, siempre que se disponga de cobertura móvil, y pueden ofrecer velocidades equiparables a las velocidades de la Banda Ancha por cable (entre 3 y 42 Mbps dependiendo del operador y del tipo de conexión (GPRS, 3G, HSDPA, HSPA+, LTE)

Para su utilización, el ordenador portátil debe integrar un módem o se debe conectar un módem externo. Las operadoras suelen ofrecer generalmente un módem bloqueado para su uso solo en esa operadora (en algunos casos subvencionado por la propia operadora) u ofrecen módem libres. Los módems externos se suelen conectar habitualmente por USB aunque al principio también se usaban por PCMCIA.

También la mayoría de los teléfonos móviles actuales soportan la función de módem. Esto le permite conectarse con un ordenador por USB o Bluetooth y (teniendo los drivers correspondientes instalados en el equipo) conectarse a internet de la misma manera que con el módem USB o PCMCIA. Aunque esto conlleva ventajas (no se necesita módem USB) también conlleva inconvenientes (como la necesidad de instalar drivers concretos del móvil, cuando con un módem USB esos controladores se instalan automáticamente). Esta forma de conexión está actualmente cayendo en desuso, principalmente debido al auge de dispositivos tipo smartphone o tableta.

³⁷ http://www.broadband.gov/spanish/broadband_types.html



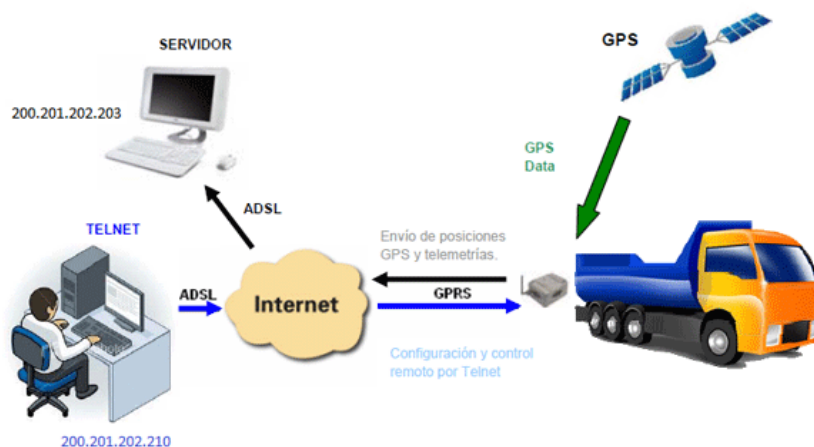
Figura 1. 19 Internet por MODEM

1.7.5.2 Para clientes comerciales e industriales

- **Líneas dedicadas T1 y T3-** Mencionamos este medio de conexión, solamente para información. Estas líneas no son para clientes domésticos, porque son muy costosas. Usualmente las usan compañías de comunicaciones. Las televisionas locales las utilizan para enviar la señal digital a los operadores de televisión por cable y a las compañías de satélites, para que los suscriptores accedan a las transmisiones locales, sin necesidad de equipo adicional. Las velocidades de esta conexión son muy altas.
- **Satélite-** Otro tipo de conexión muy rápida, pero costosa. Se accede a través de una señal inalámbrica, directamente desde un satélite, con equipo especial.
- **GPRS-** General Packet Radio Service (GPRS) o servicio general de paquetes vía radio creado en la década de los 80 es una extensión del Sistema Global para Comunicaciones Móviles (Global System for Mobile Communications o GSM) para la transmisión de datos mediante conmutación de paquetes. Existe un servicio similar para los teléfonos móviles, el sistema IS-136. Permite velocidades de transferencia de 56 a 144 kbps.

Una conexión GPRS está establecida por la referencia a su nombre del punto de acceso (APN). Con GPRS se pueden utilizar servicios como Wireless Application Protocol (WAP) , servicio de mensajes cortos (SMS), servicio de mensajería multimedia (MMS), Internet y para los servicios de comunicación, como el correo electrónico y la World Wide Web (WWW). Para fijar una conexión de GPRS para un módem inalámbrico, un usuario debe especificar un APN, opcionalmente un nombre y contraseña de usuario, y muy raramente una dirección IP, todo

proporcionado por el operador de red. La transferencia de datos de GPRS se cobra por volumen de información transmitida (en kilo o megabytes), mientras que la comunicación de datos a través de conmutación de circuitos tradicionales se factura por minuto de tiempo de conexión, independientemente de si el usuario utiliza toda la capacidad del canal o está en un estado de inactividad. Por este motivo, se considera más adecuada la conexión conmutada para servicios como la voz que requieren un ancho de banda constante durante la transmisión, mientras que los servicios de paquetes como GPRS se orientan al tráfico de datos. La tecnología GPRS como bien lo indica su nombre es un servicio (Service) orientado a radio enlaces (Radio) que da mejor rendimiento a la conmutación de paquetes (Packet) en dichos radio enlaces.



³⁸Figura 1.20 Sistema de conexión a Internet con GPRS y GPS.

³⁸ <http://www.mtxtunnel.com/blog/ejemplo-control-flotas-1-envio-de-posicion-gps-y-es-a-servidor-por-gprs/>

CAPÍTULO 2

DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE CONTROL, MONITOREO Y VIGILANCIA POR CÁMARA, VIA INTERNET, PARA LA ALIMENTACIÓN DE ANIMALES DOMESTICOS.

2.1 Especificación del Diseño (Introducción).

Este capítulo contiene toda la información del diseño tanto del hardware como del software del sistema. En la primera parte se analizan los elementos, las cuales serán sujetas al monitoreo y el respectivo control. Para el diseño del hardware se incluye información sobre cada etapa diseñada y además las especificaciones técnicas de acuerdo a las necesidades presentadas durante el diseño del sistema, así mismo se explica el desarrollo del programa que permita la interface entre el usuario y el sistema el cual fue desarrollado con la tecnología Raspberry Pi, misma que permite de manera cómoda la comunicación del usuario con el sistema. Además se explica la configuración de la computadora que se enlaza al Internet, para la comunicación con el Raspberry Pi para que pueda monitorear y controlar los cambios de las variables en tiempo real. Finalmente se presenta el montaje del sistema completo.

2.2 Análisis de la Estructura.

La estructura estará compuesta de dos recipientes que suministrarán el alimento sólido y líquido al animal, cuando el usuario se conecte al Internet y controle y vigile el sistema.



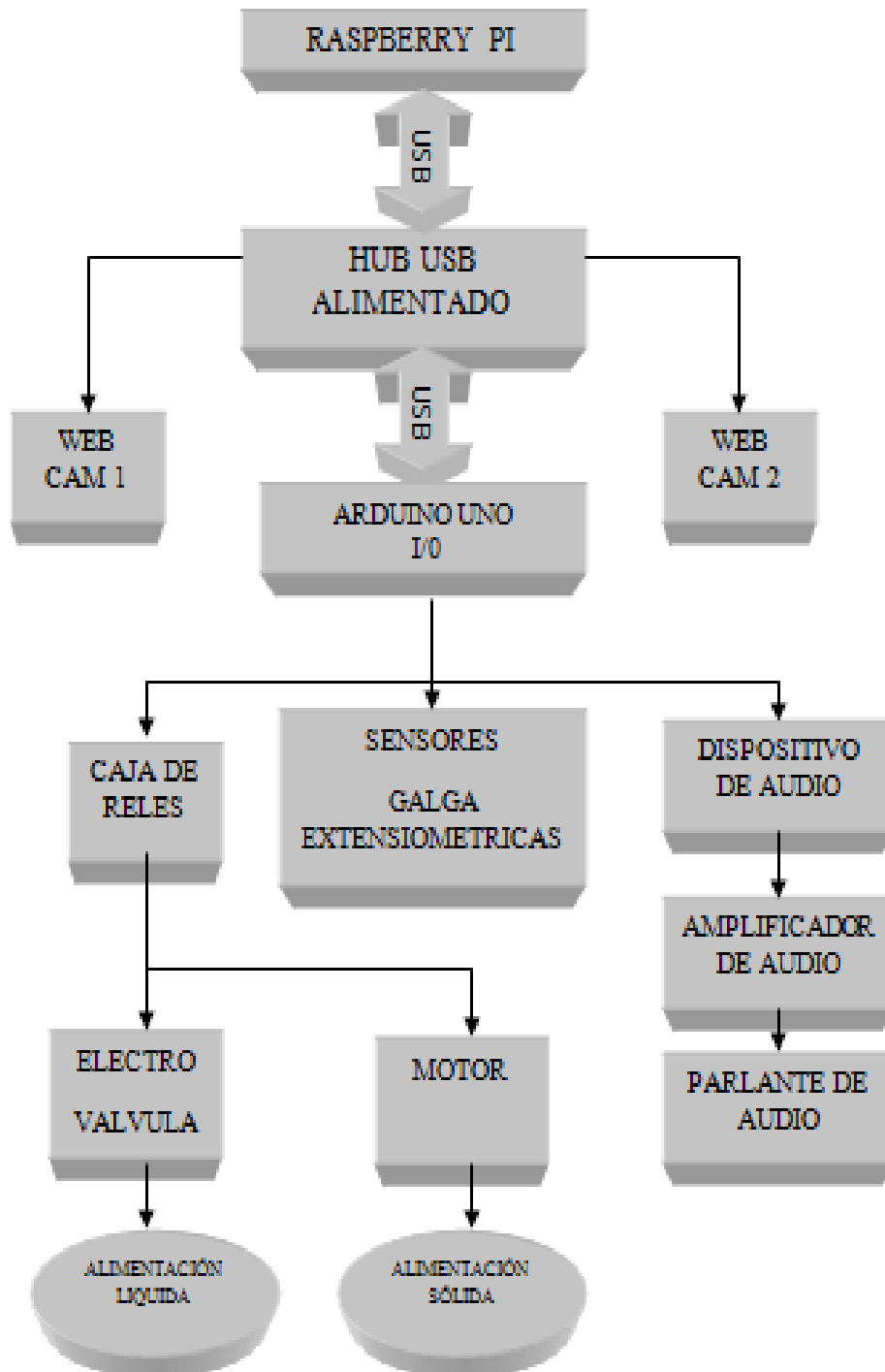


Figura 2.1: Diagrama de bloques del control, monitoreo y vigilancia de canes.

2.2.1 Operación de la Estructura.

La operación de la estructura será manipulada, para la alimentación líquida, el agua será por una electroválvula que conducirá a un recipiente hasta un determinado nivel que será programado, así el Sensor enviara una señal para que se cierre la electroválvulas y termine adecuadamente el proceso.

Para la alimentación sólida será manipulada mediante una tolva y esta será trasportada mediante una banda trasportadora al recipiente, el cual se censará el nivel de alimentación para terminar el proceso.

2.2.2 Métodos de control de la Estructura.

Para controlar la estructura se realiza de dos formas:

- La primera de forma remota desde el propio Raspberry Pi.
- La segunda desde el Internet.

Para el proyecto propuesto se emplea el método de control desde el Internet.

2.2.2.1 Método de forma remota desde el propio Raspberry Pi.

Este método es mucho más fácil ya que no necesita la conexión a Internet, solo basta tener el Raspberry Pi funcionando y conectado a los elementos de control de la estructura para tener un funcionamiento óptimo.

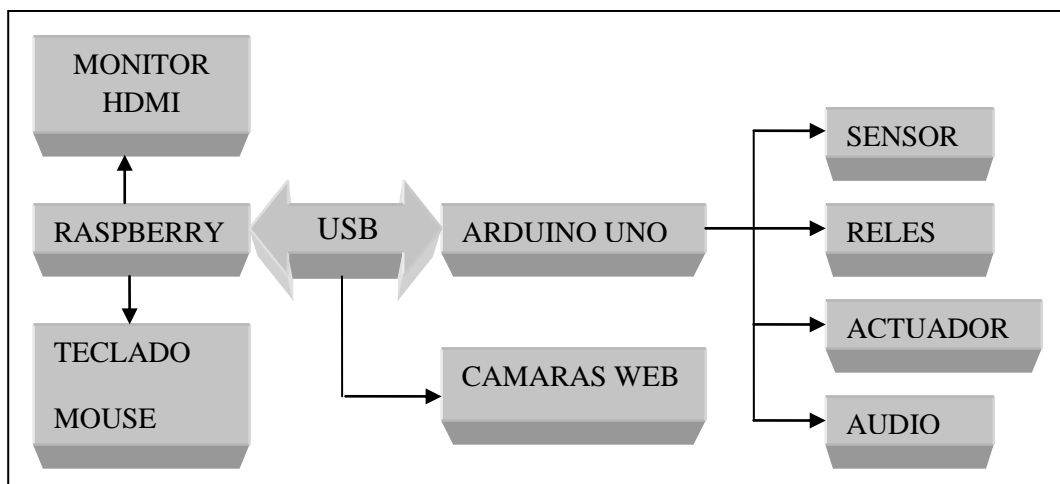


Figura 2.2: Etapas del hardware del sistema sin Internet

2.2.2.2 Método de control mediante el Internet.

El método de control desde el Internet es en la cual nos vamos a enfocar para la manipulación del sistema, puesto que para poder controlar y monitorear desde el Internet los dispositivos de control se necesitara que el Raspberry Pi sea manipulado por otra computadora en forma de Conexión de Escritorio Remoto y esta computadora sea la que enlace al Internet el Raspberry para ser controlado y manipulado por el usuario gracias a un software gratis la cual nos permite comunicar desde cualquier parte donde nos encontremos.

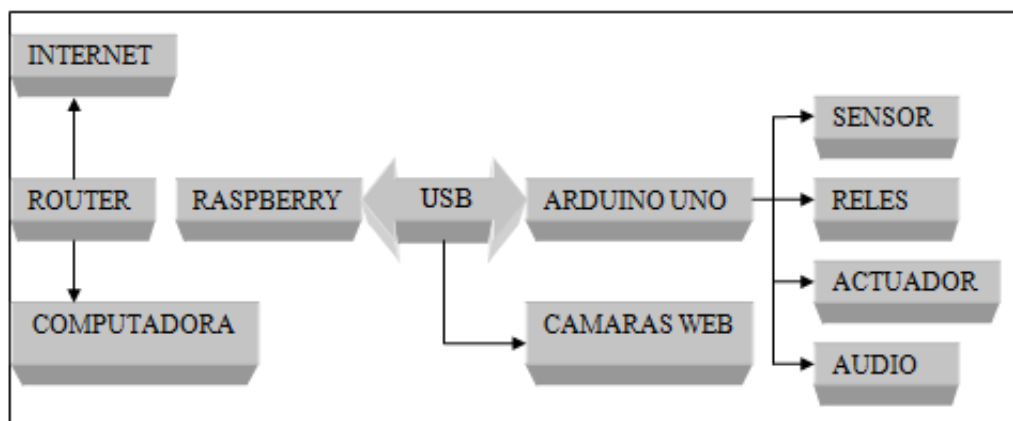


Figura 2.3: Etapas del hardware del sistema con Internet

2.3 Diseño del SOFTWARE.

En esta sección se referirá al diseño y características de los diferentes paneles frontales que servirán de enlace entre el usuario y el control del proceso en general.

2.3.1 Implementación del Software

2.3.1.1 Programación en Raspberry Pi con el Software PYTHON.

Python es una plataforma de programación orientada a objetos que posee Raspberry Pi, por lo tanto la persona que desarrolla una aplicación en este software, posee muchas herramientas que facilitan, el diseño, la construcción y desempeño, del mismo.

El programa realizado en Python se indica en Anexos.

Ingresamos al escritorio del Raspberry Pi, posteriormente abrimos el programa de alimentacióncan.py en el escritorio donde correremos el programa haciendo doble clic en el mismo:

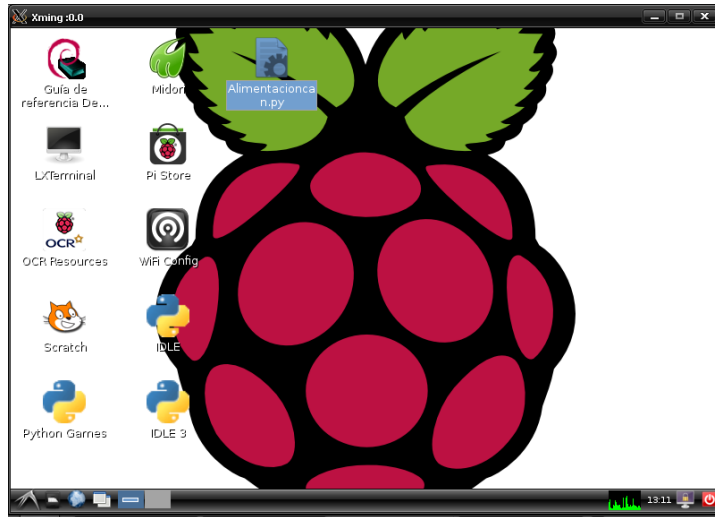


Figura 2.4: Visualización del Escritorio del Raspberry Pi.

Al realizar doble clic en nuestro programa alimentacióncan.py nos indicara una ventana donde escogeremos la opción EJECUTAR y se abrirá la ventana con nuestro panel principal.



Figura 2.5: Ventana donde se escoge la opción Ejecutar del programa de alimentación.

Cuando el programa de Python este corriendo se aparecerá una nueva ventana con los botones que controlan el proceso de alimentación.



Figura 2.6: Ventana del proceso de alimentación.

2.3.1.2 Programación en Arduino Uno.

Arduino uno es una interface con la comunicación con el Raspberry Pi.

El Arduino es la interfaz con los elementos de control de entrada y salida.

```

Alimentacioncan
int i = 0;
float val = 0;
float vall = 1;

#define leduno 3
#define leddos 4
#define ledtres 5
#define ledcuatro 6
#define play 7
#define next 8
#define reset 9
#define retroceder 10
void setup() {
  Serial.begin(9600);
  for (i=3; i<=10; i++){
    pinMode(i, OUTPUT);
  }
  pinMode(12, OUTPUT);
  pinMode(13, OUTPUT);
}
void loop() {
  if(Serial.available())
  
```

Figura 2.7: Ventana que contiene el programa realizado en Arduino.

Nota: El programa correspondientemente realizado esta en el Anexo2.

2.3.1.3 Panel Frontal Principal Este panel permite al usuario seleccionar Los botones de los respectivos procesos a controlar o a su vez alguna operación específica que desee realizar.

El panel principal constan los siguientes Botones del proceso de alimentación.



Figura 2.8: Panel frontal del proceso de alimentación.

A continuación se detalla cada uno de los botones presentes en este panel de control.

- **Alimentación Sólida:**

En esta parte encontramos el botón de “Alimentación Sólida” y un aviso que nos indica que debemos presionar el botón de alimentación sólida “PRESIONE*BOTON*S”, para que este entre en funcionamiento y además tenemos un indicador luminoso que cuando está en rojo nos indica que el proceso está apagado.

Si se escoge la opción “Alimentación sólida” y se pulsa el botón mencionado, estamos enviando una señal para que se active el motor de la banda transportadora de comida y este dirija el alimento al recipiente adecuado.

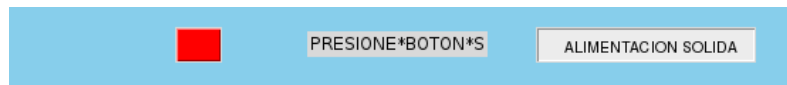


Figura 2.9: Botones en estado de Apagado de la Alimentación Sólida.

Cuando se ha realizado la activación del botón de alimentación sólida, el Sensor entrara en funcionamiento hasta que este detecte el peso deseado de la alimentación y terminado el proceso, el aviso se pone en ****FINALIZADO**** y el indicador de luminosidad se activa de color verde con este proceso se ha indicado que la alimentación sólida ha terminado.



Figura 2.10: Botones en estado de Encendido de la Alimentación Sólida

• Alimentación Líquida:

En esta parte encontramos el botón de “Alimentación Líquida” y un aviso que nos indica que debemos presionar el botón de alimentación líquida “PRESIONE*BOTON*L”, para que este entre en funcionamiento y además tenemos un indicador luminoso que cuando está en rojo nos indica que el proceso está apagado.

Si se escoge la opción “Alimentación Líquida” y se pulsa en el botón, estamos enviando una señal para que se active la electroválvula y este dirija el alimento líquido al recipiente adecuado.



Figura 2.11: Botones en estado de Apagado de la Alimentación Líquida.

Cuando se ha realizado la activación del botón de alimentación líquida, el Sensor entrara en funcionamiento hasta que este detecte el peso deseado de la alimentación líquida y terminado el proceso, el aviso se pone en

****FINALIZADO**** de color verde como el indicador luminoso que también se pone de color verde.



Figura 2.12: Botones en estado de Encendido de la Alimentación Líquida.

• **Luz de los Alimentos:**

Con el siguiente botón podemos brindar luz en la parte de alimentación en caso de que el usuario tenga que observar y brindar la alimentación a sus animales en la noche.

La activación está compuesta por un botón que dice “LUZ DE LOS ALIMENTOS” y el apagado del mismo está por otro botón que dice “APAGAR LUZ PERRO”, un aviso que nos indica “LUZ APAGADA” y un indicador luminoso que esta de color rojo cuando está apagado



Figura 2.13: Botones en estado de Apagado de la Luz de los Alimentos.

Cuando el proceso esta encendido se indicara en palabras “ENCENDIDO” y el indicador luminoso se mostrara de color verde, posteriormente se apagara con el botón “APAGAR LUZ PERRO”.



Figura 2.14: Botones en estado de Encendido de la Luz de los Alimentos.

• **Luz de Exteriores:**

Con el siguiente botón podemos brindar luz en la parte de los exteriores en caso de que el usuario tenga que observar y vigilar mediante la cámara a sus animales en la noche.

La activación está compuesta por un botón que dice “LUZ DE EXTERIORES” y el apagado del mismo está por otro botón que dice “APAGAR LUZ EXTERIOR”, un aviso que nos indica “LUZ APAGADA” y un indicador luminoso que esta de color rojo cuando está apagado



Figura 2.15: Botones en estado de Apagado de la Luz de Exteriores.

Cuando el proceso esta encendido se indicara en palabras “ENCENDIDO” y el indicador luminoso se mostrara de color verde, posteriormente se apagara con el botón “APAGAR LUZ EXTERIORES”.



Figura 2.16: Botones en estado de Encendido de la Luz de Exteriores.

- **Control de audio para el llamado del animal:**

Con los siguientes botones se puede hacer reproducir los diferentes tonos o llamados al animal cuando estos estén pre grabados en el dispositivo de audio wtv020sd16p.



Figura 2.17: Botones de audio.

- **Control de registro de consumo y verificación en la alimentación del can.**

Con los siguientes botones realizamos una verificación del consumo del alimento suministrado al can y podemos saber cuánto de comida nos resta del total de la comida y los datos se quedan guardados en un registro para posteriores consulta.



Figura 2.18: Botones de registro de alimentos.

En la siguiente figura 2.19 se muestra el lugar donde se encuentran los registros guardados para posteriores consultas.

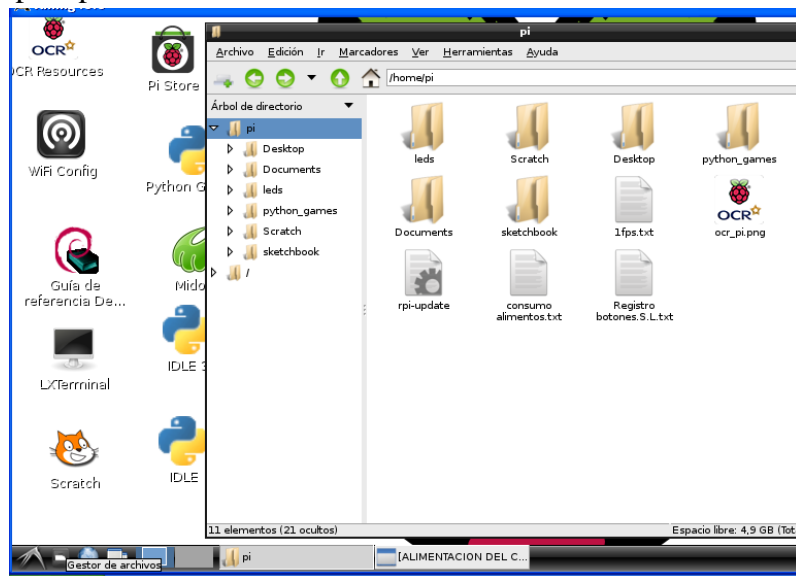


Figura 2.19: Ventana donde se encuentran los registros de consumo de los alimentos.

2.3.2 Programación en Raspberry Pi con el Software HTML Y PHP5 para las 2 cámaras Web.

Pasos para instalar las cámaras Web.

Ahora sí, instalamos el paquete Motion mediante el comando

- `sudo apt-get install motion`

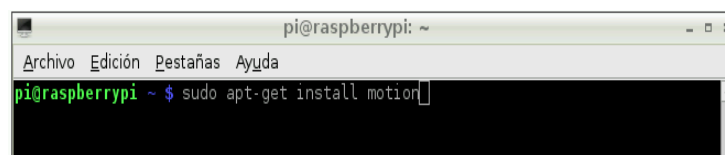


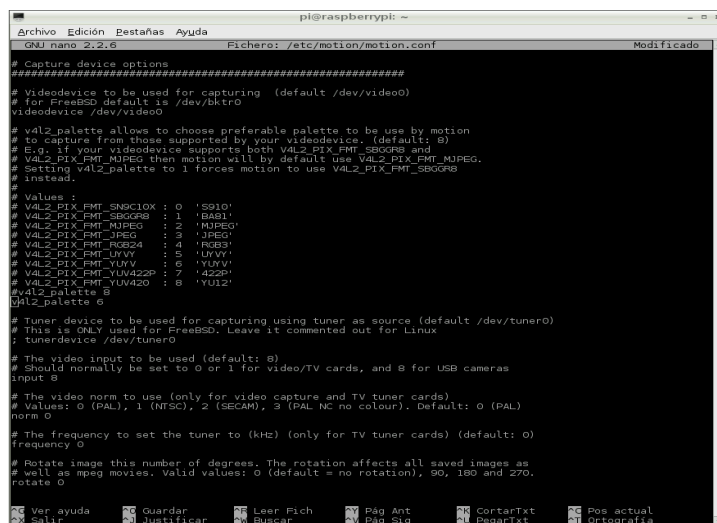
Figura 2.20: Ventana para instalar en programa motion de las cámaras.

Con el siguiente comando verificamos si se encuentran conectadas las dos cámaras:

- Ls /dev/vid*

Ahora procedemos a ingresar al comando motion para configurar las cámaras web:

- Sudo nano /etc/motion/motion.conf



```
pi@raspberrypi: ~
GNU nano 2.2.6 Fichero: /etc/motion/motion.conf Modificado
# Capture device options
#####
# Videodevice to be used for capturing (default /dev/video0)
# for FreeBSD default is /dev/bktr0
videodevice /dev/video0
# v4l2_palette allows to choose preferable palette to be use by motion
# to capture from those supported by your videodevice. (default: 8)
# E.g. if your videodevice supports both V4L2_PIX_FMT_SBGGR8 and
# V4L2_PIX_FMT_MJPEG then motion will by default use V4L2_PIX_FMT_MJPEG.
# Setting v4l2_palette to 1 forces motion to use V4L2_PIX_FMT_SBGGR8
# instead.
#
# Values:
# V4L2_PIX_FMT_SBGGR8 : 0 'S910'
# V4L2_PIX_FMT_SBGGR8 : 1 'BA81'
# V4L2_PIX_FMT_MJPEG : 2 'MJPEG'
# V4L2_PIX_FMT_MJPEG : 3 'JPEG'
# V4L2_PIX_FMT_RGB564 : 4 'RGB564'
# V4L2_PIX_FMT_UYVY : 5 'UYVY'
# V4L2_PIX_FMT_YUV420 : 6 'YUV420'
# V4L2_PIX_FMT_YUV422 : 7 'YUV422'
# V4L2_PIX_FMT_YUV420 : 8 'YU12'
# V4L2_palette 8
V4L2_palette 8
# Tuner device to be used for capturing using tuner as source (default /dev/tuner0)
# This is ONLY used for FreeBSD. Leave it commented out for Linux
; tunerdevice /dev/tuner0
# The video input to be used (default: 8)
# Should normally be set to 0 or 1 for video/TV cards, and 8 for USB cameras
input 8
# The video norm to use (only for video capture and TV tuner cards)
# Values: 0 (PAL), 1 (NTSC), 2 (SECAM), 3 (PAL_NC no colour). Default: 0 (PAL)
norm 0
# The frequency to set the tuner to (kHz) (only for TV tuner cards) (default: 0)
frequency 0
# Rotate image this number of degrees. The rotation affects all saved images as
# well as mpeg movies. Valid values: 0 (default = no rotation), 90, 180 and 270.
rotate 0
Ver ayuda Guardar Leer Fich Pág Ant CortarTxt Pos actual
Salir Justificar Buscar Pág Sig PegarTxt Ortografía
```

Figura 2.21: Ventana de configuración de motion.

- En zona **## Daemon ##**
 - > En el apartado **#Daemon#** se cambia **Daemon OFF** por **Daemon ON**
 - > Nota: su utilidad es obvia.
- En zona **## Live Webcam Server ##**
 - > **webcam_port 8001**
 - > **webcam_localhost off**
 - > Nota: habilita puerto de visualización y la misma en otros equipos de la red.
- En zona **## HTTP Based Control ##**
 - > **control_port 8888**
 - > **control_localhost off**
 - > Nota: habilita puerto de configuración y el acceso al control desde otros equipos de la red.

- En zona `## Thread config files ##` (al final del fichero)
 - > Therad conf files
 - > thread /etc/motion/cam1.conf
 - > thread /etc/motion/cam2.conf

```

pi@raspberrypi: ~
Archivo Edición Pestañas Ayuda
GNU nano 2.2.6 Fichero: /etc/motion/motion.conf
# Only put the options that are unique to each camera in the
# thread config files.
; thread /usr/local/etc/thread1.conf
; thread /usr/local/etc/thread2.conf
; thread /usr/local/etc/thread3.conf
; thread /usr/local/etc/thread4.conf

thread /etc/motion/cam1.conf
thread /etc/motion/cam2.conf
  
```

Figura 2.22: Ventana de configuración de motion.conf.

Nota: Añade un fichero de configuración específico para nuestra cámara.

- Ahora, y como configuración extra, creamos los fichero cam1.conf y cam2.conf mediante lo siguiente:
- Fichero cam1.conf:
 - > sudo nano /etc/motion/cam1.conf
 - > Y dentro del mismo editamos creando el contenido:
 - > videodevice /dev/video0
 - > webcam_port 8081
- Como siempre, salvaremos con `Ctrl-X + S + <Enter>`

```

pi@raspberrypi: ~
Archivo Edición Pestañas Ayuda
GNU nano 2.2.6 Fichero: /etc/motion/cam1.conf
videodevice /dev/video0
webcam_port 8081
  
```

Figura 2.23: Ventana de configuración para la cámara 1.

- Fichero cam2.conf
 - > sudo nano /etc/motion/cam2.conf

> Y dentro del mismo editamos creando el contenido:

> videodevice /dev/video1

> webcam_port 8082

- Como siempre, salvaremos con **Ctrl-X + S + <Enter>**

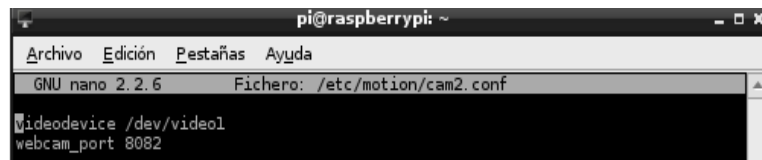


Figura 2.24: Ventana de configuración para la cámara 2.

El fichero cam1.conf y cam2.conf ha de residir en el mismo directorio donde se encuentra motion.conf y para verificar esto tecleamos el siguiente comando `ls -l /etc/motion/` y se puede observar claramente que están reconocidas las dos cámaras que hemos introducido para el sistema de vigilancia.

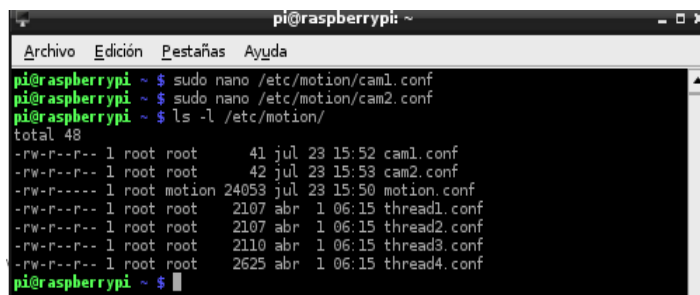


Figura 2.25: Ventana de verificación de instalación de las dos cámaras cam1 y cam2.

> Abrimos el segundo fichero de configuración:

- Sudo nano /etc/default/motion
 - > Cambiamos el parámetro siguiente:
 - > Start_motion_daemon de NO a YES

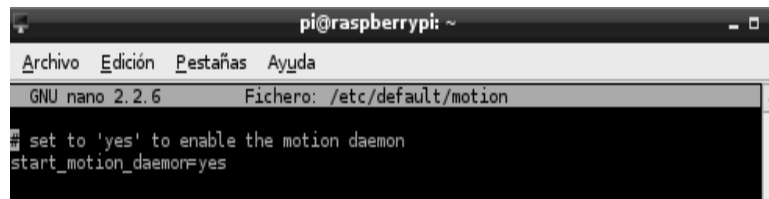


Figura 2.26: Ventana de configuración de motion el auto inicio de las cámaras.

- Iniciamos la emisión de video.

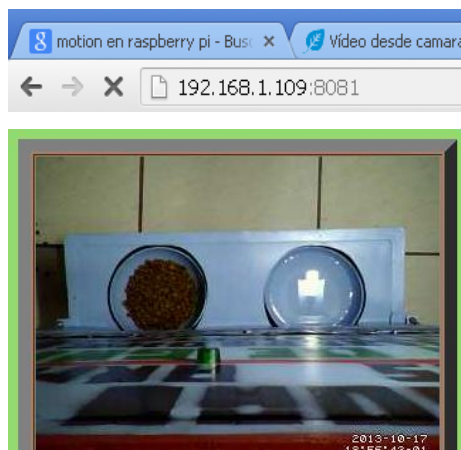
Como configuramos ponemos:

Sudo service motion restart.

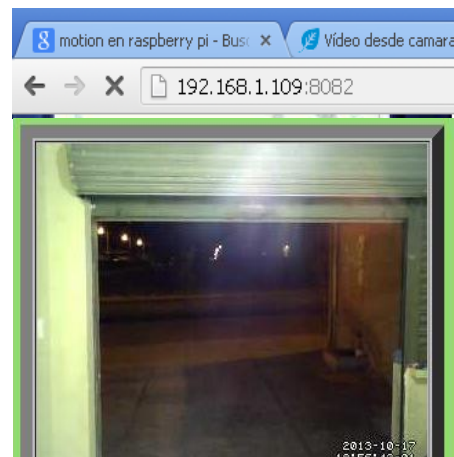
2.3.2.1 Ahora ya podemos acceder a ver las Webcams desde la dirección indicada.

Cámara 1 es: 192.168.1.109:8081

Cámara 2 es: 192.168.1.109:8082



a) Cámara 1



b) Cámara 2

Figura 2.27: Cámaras establecidas en su dirección adecuada.

2.3.2.2 Automatización de arranque y parada de motion.

- Fichero de activación.
 - > Sudo nano /usr/bin/onmotion y tecleamos dentro del mismo:
 - > Sudo motion
- Fichero de parada
 - > Sudo nano /usr/bin/offmotion y tecleamos dentro del mismo:

- > Sudo /etc/init.d/motion stop
- En ambos casos para guardar y salir ctrl-O y para salir ctrl-X Enter
- Para poder dar permisos a las carpetas.
 - > Sudo chmod 777 /usr/bin/onmotion
 - > Sudo chmod 777 /usr/bin/offmotion
- A partir de este momento para arrancar las cámaras tecleamos:
 - > Onmotion para iniciar y para finalizar tecleamos offmotion dentro del Híper terminal.

2.3.2.3 Ahora realizaremos la unión de las dos cámaras.

Realizamos el siguiente comando sudo nano /var/www/camarasweb.html para teclear dentro de este el siguiente código realizado en HTML que el mismo se encuentra en Anexo3.

Cuando se ha finalizado el proceso de programación de las dos cámaras se introduce la dirección 192.168.1.109/camarasweb.html para observar en tiempo real el proceso de alimentación y la vigilancia de los animales.

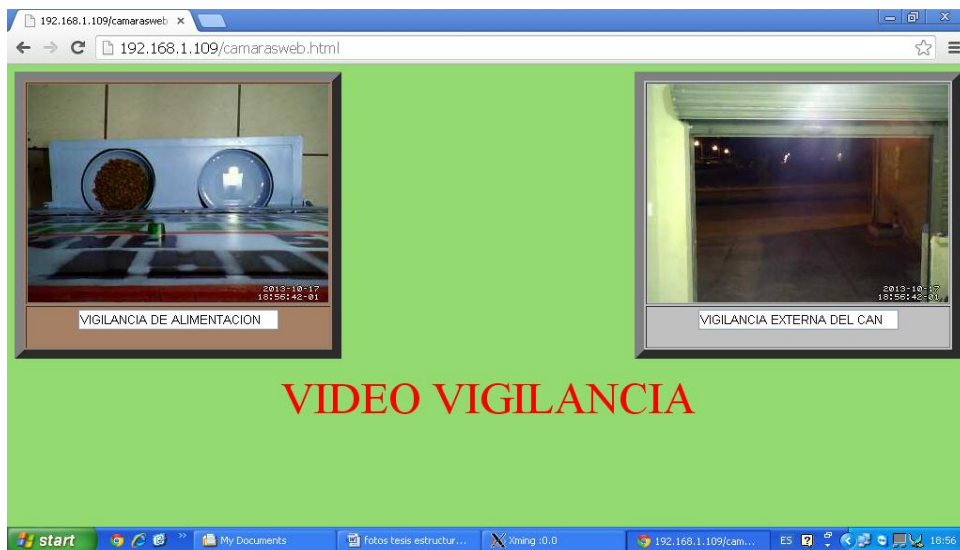


Figura 2.28: Ventana donde se muestra las dos cámaras con su dirección establecida.

2.4 Diseño del HARDWARE.

Para el diseño del hardware o parte física se basa en una estructura donde se encuentran los recipientes de alimentación, en donde van colocados los distintos sensores y electroválvulas.



Figura 2.29: Estructura frontal del sistema de alimentación

2.4.1 Implementación del Hardware.

2.4.1.1 Internet Por Modem 3g USB De Claro

Banda Ancha Móvil es un servicio que te permite disfrutar de una conexión a Internet de manera inalámbrica y rápida, navegando libremente desde tu Laptop o PC mediante la red 3.5G de Claro.

- Funcionalidades:

- Conexión a Internet.
- Acceso a correo electrónico.
- Envío y transferencia de archivos.
- Monitoreo de sistemas remotos.
- M-commerce y M-banking.
- MP3, Videos, Chat, y otros entrenamientos.

- **Beneficios:**

- **Movilidad:** Permite el acceso desde cualquier punto en cualquier momento incluso donde no existe ningún tipo de comunicación convencional.
- **Simplicidad en el acceso:** Mediante MODEMS de fácil instalación.
- **Comodidad:** Permite la navegación desde cualquier lugar que se encuentre dentro de cobertura.
- **Cobertura:** Ver Cobertura 3.5G.
- **Velocidad:** Navegar a gran velocidad con Tecnología 3.5G.

2.4.1.2 Router. Comparte la libertad de 3G/4G Alimentado por computadora portátil o adaptador de corriente, el TL-MR3020 permite al usuario compartir de manera sencilla una conexión móvil 3G/4G con familia y amigos en el tren, acampando, en el hotel, y casi en cualquier lugar por su cobertura 3G/ 4G. Al conectar un modem USB 3G/4G al router, un hotspot 3G/4G con Wi-Fi se establece de manera inmediata.

El TL-MR3020 se puede instalar rápida y sencillamente en casi cualquier lugar. Solo necesita insertar el modem USB en el router, buscar el SSID inalámbrico en su computadora y conectarse a este. Los usuarios pueden estar en línea y navegando la red en instantes, utilizando la guía de usuario provista en disco compacto, incluso los usuarios novatos pueden establecer de manera sencilla sus conexiones de Internet inalámbrico.

2.4.1.3 La Computadora. En este sistema es el medio para interactuar con el Raspberry Pi mediante una conexión de Acceso Remoto.

2.4.1.4 Raspberry Pi. Es el cerebro del sistema tiene la interfaz con el usuario, en el que se ha instalado Linux, que es compatible con PYTHON IDLE, en donde se ha desarrollado la aplicación, y cumple además las funciones de servidor web. Para comunicación del computador con elementos externos se está utilizando el puerto usb, para la comunicación con el Arduino uno, ya que permite obtener una velocidad aceptable, fácil programación, accesible y muy práctico.

2.4.1.5 Módulo I/O

2.4.1.5.1 Hardware Del Arduino Uno

Al ser Arduino una plataforma de hardware libre tanto su diseño como su distribución puede utilizarse libremente para el desarrollo de cualquier tipo de proyecto sin haber adquirido ninguna licencia. Por eso existen varios tipos de placa oficiales, las creadas por la comunidad Arduino o las no oficiales creadas por terceros pero con características similares. En la placa Arduino es donde conectaremos los sensores, actuadores y otros elementos necesarios para comunicarnos con el sistema.



a) Parte Frontal

b) Parte Posterior

Figura 2.30: Forma Física del Arduino uno

Es el último modelo diseñado y distribuido por la comunidad Arduino. La placa tiene un tamaño de 75x53mm. Su unidad de procesamiento consiste en un microcontrolador Atmega328. Puede ser alimentada mediante USB o alimentación externa y contiene pines tanto analógicos como digitales.

A continuación se muestra en la Ilustración donde están ubicados los elementos más importantes que componen la placa Arduino Uno que son descritos de arriba abajo y de izquierda a derecha:

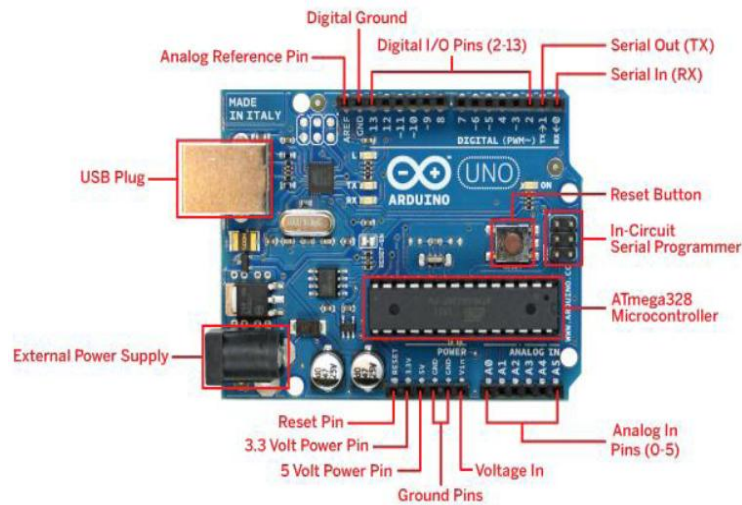


Figura 2.31: Arduino uno y sus elementos más importantes de su placa.

2.4.1.5.2 Referencia para pines analógicos (AREF)

Tensión de referencia para entradas analógicas. Se utiliza con la función `analogReference()`.

• Pines digitales de entrada y salida

En estos pines conectaremos la patilla de dato del Sensor/actuador. Desde ellos podremos leer la información del Sensor o activar el actuador. Hay 14 pines digitales que pueden utilizarse como entrada o salida con las funciones `pinMode()`, `digitalWrite()`, y `digitalRead()`. Operan a 5 voltios. Cada pin proporciona o recibe como máximo 40mA y disponen de una resistencia pull-up (desconectada por defecto) de 20-50 kOhmios. Ciertos pines son reservados para determinados usos: Serie: 0(RX) y 1(TX). Utilizados para recibir (RX) y transmitir (TX) datos serie. Están directamente conectados a los pines serie del microcontrolador. Utilizando estos pines podremos conectarnos con otras placas.

Interrupciones externas: 2 y 3. Estos pines pueden ser configurados para activar interrupciones.

PWM: 3, 5, 6, 9, 10 y 11. Proporcionan una salida de 8 bits en modo PWM.

SPI: 10-13. Estos pines soportan la librería de comunicación de dispositivos SPI.

LED: 13. Este pin está conectado con un led de la placa. Cuando se le asigne un valor HIGH se encenderá, en cambio si lo dejamos en LOW estará apagado.

- **Pines analógicos**

Esta placa contiene 6 pines de entrada analógicos. Los elementos que se conecten aquí suelen tener mayor precisión que los digitales pero su uso requiere de una lógica levemente mayor. Más adelante se comentará el uso de un termistor analógico.

2.4.1.6 Caja De Relés Las salidas del módulo I/O son manejadas por relés de 5 voltios, que envían una señal de 12Vcd y una de 110 voltios de baja corriente a la caja de relés, donde estos son los encargados del control de las cargas, sea iluminación, motor o electroválvulas de la estructura de alimentación. Estos relés tienen dos o tres contactos que permiten el control simultáneo por zonas tal como se pretende en el sistema.

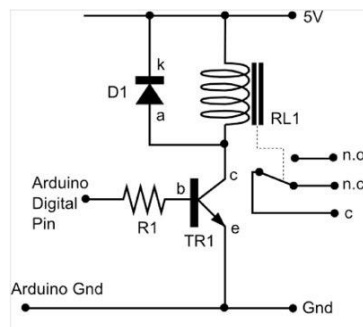


Figura 2.32: Diseño de la entrada y salida del relé para el manejo del actuador.

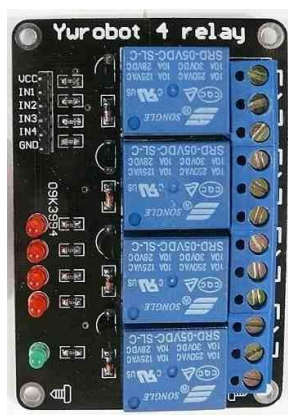


Figura 2.33: Forma física los cuatro relés que se conectaran a los actuadores.

2.4.1.7 Sensores Y Carga

Las cargas son compuestas principalmente por el sistema de alimentación como una electroválvula, motor y sistema de iluminación. A estos se los ha dividido en zonas para su administración remota. Cada uno se los maneja en forma independiente y con la jerarquía de que cuando se activa el sistema remoto no se pueda hacer ninguna acción en forma manual.

Por tanto la corriente de la fuente debe ser diseñada para suministrar mínimo 2A y 12V a las cargas. En la parte práctica se diseña con un transformador de 110V / 12V a 2A. Un puente rectificador de 8A, un condensador o filtro de 2200uF/25V y para tener una señal de voltaje constante un circuito integrado regulador L7812CV como se muestra en la figura 2.32:

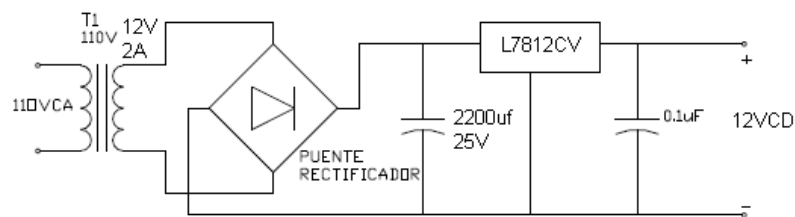


Figura 2.34: Fuente de alimentación para los módulos de carga.

El Sensor a utilizar para el ingreso de las señales es de tipo peso como la galga extensiométrica de cuatro hilos, dos de alimentación y dos de la señal de salida, que nos da los valores deseados al ponerle peso.

Son utilizados por ser casi imperceptibles, silenciosos y de fácil adquisición.



a) Sensor (Galga extensiométrica)



b) Electroválvula del Proceso

Figura 2.35: Sensor y actuador del Proceso

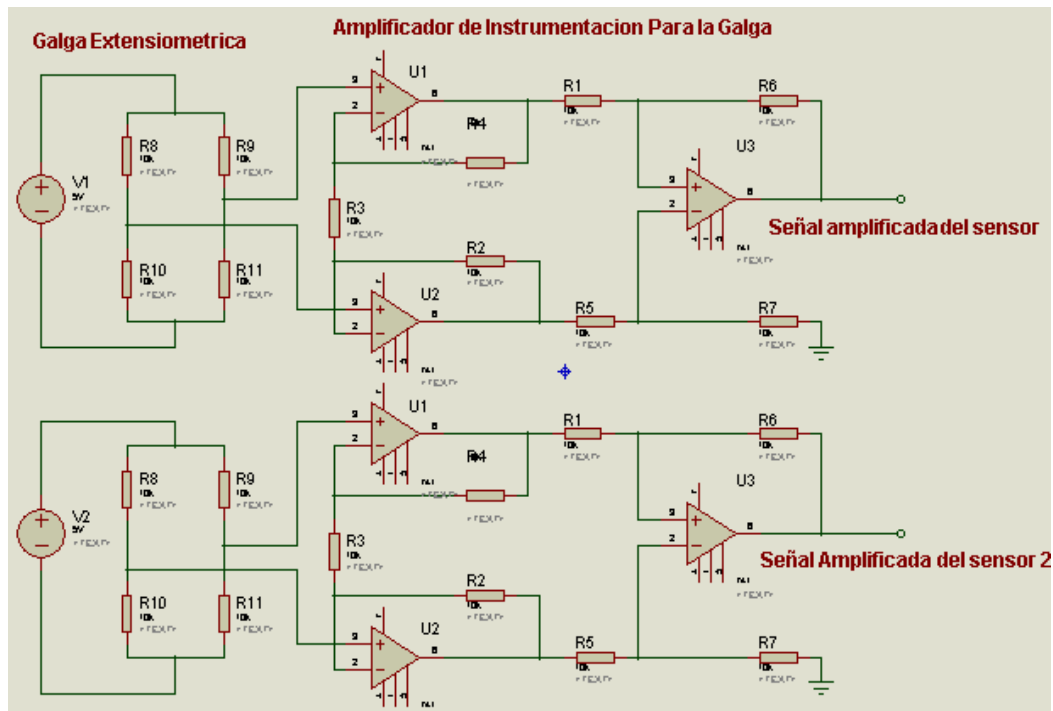


Figura 2.36: Acondicionamiento de señal del Sensor (Galga extensiométrico)

2.4.1.8 Dispositivo De Audio

Con el fin de que el usuario pueda llamar a su mascota para que se alimente, se ha implementado un sistema de audio que será activado desde el panel principal del control y monitoreo del sistema.

Para la implementación de este sistema de audio se ha invertido en los diferentes dispositivos electrónicos que son:

- Decodificador wtv020-sd-16p
- Modulo amplificador de audio pam8403 de 3w

2.4.1.8.1 Decodificador Wtv020-Sd-16p

Los componentes empiezan a llegar, el primero ha sido el decodificador de ficheros de música WTV020-SD-16P es un decodificador de música WTV020 con tarjeta MicroSD de 16 Pins de salida.

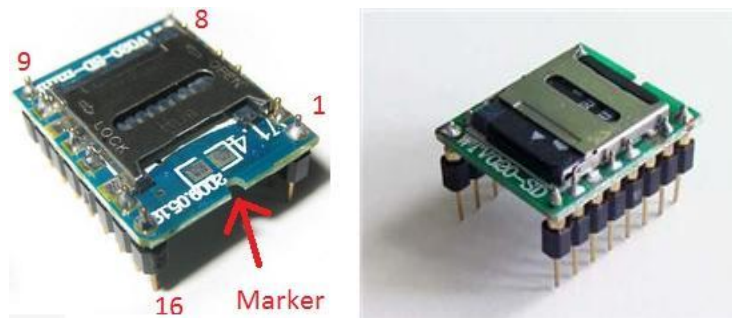


Figura 2.37: Panel frontal del dispositivo de reproducción y su distribución de pines con su tarjeta SD.

Lo primero que puedo observar es su reducido tamaño, es un poco mayor que el tamaño de una MicroSD. Precisamente esta es la primera pega que hay que salvar, encontrar una tarjeta MicroSD compatible, por lo que he podido leer por ahí acepta hasta tarjetas de 2GB pero de ciertas marcas, en mi caso las que yo tengo de 2GB no son compatibles pero una Scan disk de 1GB sí que funciona. Por lo tanto voy a utilizar una tarjeta San disk de 1GB.

| | | | |
|---|---------|-----|----|
| 1 | RESET | VDD | 16 |
| 2 | AUDIO-L | P06 | 15 |
| 3 | NC | NC | 14 |
| 4 | SPK+ | P02 | 13 |
| 5 | SPK- | P03 | 12 |
| 6 | NC | NC | 11 |
| 7 | P04 | P05 | 10 |
| 8 | GND | P07 | 9 |

| PIN | SYS. | DESCRIPTION | FUNCTION |
|-----|---------|-------------|----------------------------------|
| 1 | RESET | RESET | Reset pin |
| 2 | AUDIO-L | DAC+ | DAC audio output(+) to amplifier |
| 3 | NC | NC | NC |
| 4 | SP+ | PWM+ | PWM audio output to speaker |
| 5 | SP- | PWM- | PWM audio output to speaker |
| 6 | NC | NC | NC |
| 7 | P04 | K3/A2/CLK | Key /CLK in two line serial |
| 8 | GND | GND | Address pin |
| 9 | P07 | K5/A4/SBT | Key |
| 10 | P05 | K4/A3/DI | Key /DI in two line serial |
| 11 | NC | NC | NC |
| 12 | P03 | K2/A1 | Key |
| 13 | P02 | K1/A0 | Key |
| 14 | NC | NC | NC |
| 15 | P06 | BUSY | BUSY pin |
| 16 | VDD | VDD | Power input |

Figura 2.38: Descripción de pines del módulo de reproducción de audio WTV020-SD-16P

Para hacer este testeo en la MicroSD metemos algunos sonidos de prueba en formato AD4 un formato especial.

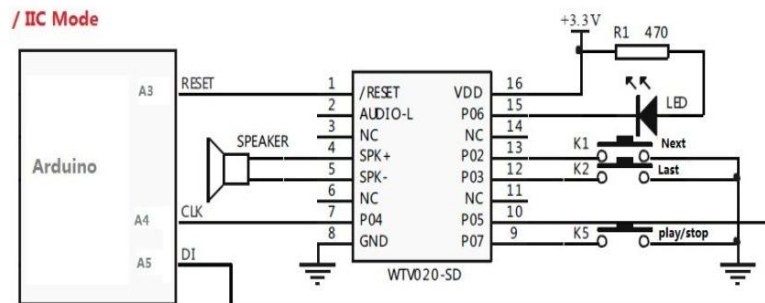


Figura 2.39: Conexiones para la reproducción de audio

2.5 Servicio de Internet proporcionado por distribuidor de Internet.

Una vez ya desarrollado y comprobado el funcionamiento del sistema en la casa, es necesario adaptarlo y levantar el programa con ayuda de las herramientas de TeamViewer en forma de Acceso Remoto.

La computadora que estará conectada al Raspberry Pi de forma remota, debe estar inicializado el programa TeamViewwer para que tenga una conexión exitosa.

Una vez instalado el programa TeamViewer, desde el lugar que vayamos a establecer una comunicación con la otra computadora que estará conectado al Raspberry Pi de forma remota procedemos a correr el programa TeamViewer.

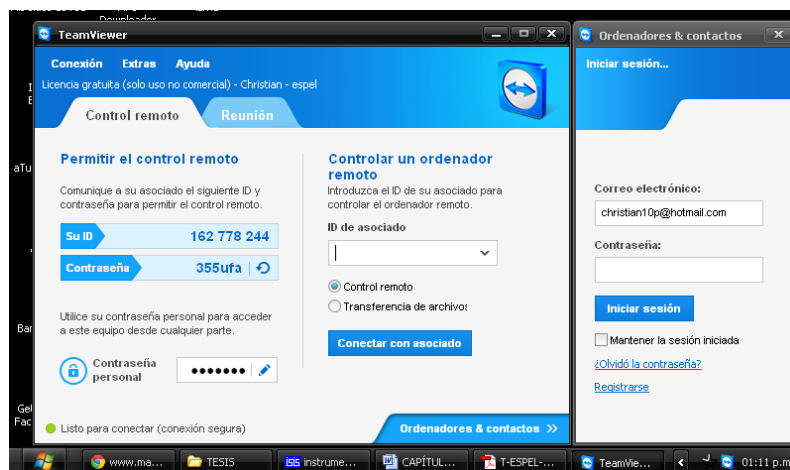


Figura 2.40: Ventana del programa TeamViewer.

Primero introducimos su ID de asociado 867102465 y la contraseña personal establecida *espel2013* y procedemos a conectar con la otra computadora de forma remota.

Una vez establecido la conexión con la otra computadora y obtener el escritorio del mismo procedemos a buscar el programa ifconfig que se encuentra en el escritorio.



Figura 2.41: Corrido del programa TeamViewer en Escritorio Remoto.

Procedemos a correr el programa ifconfig para obtener el siguiente cuadro, donde se introducira la contraseña *espel2013* y proceder hacer un escritorio remoto con el raspberry pi mediante el programa Xming.

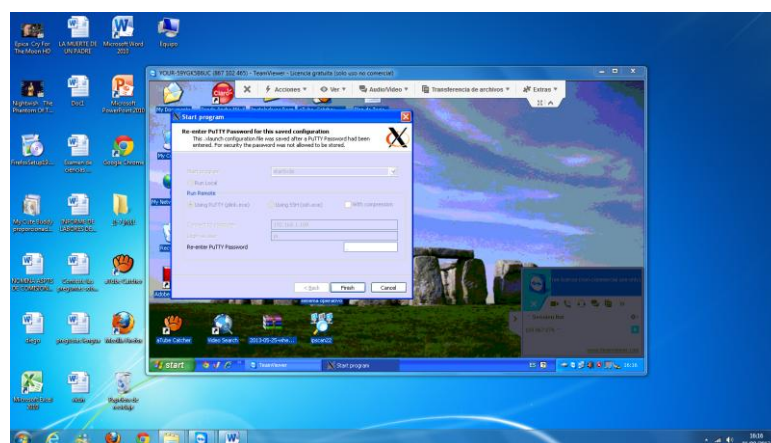


Figura 2.42: Corrido del programa Xming para acceder al Escritorio Remoto del Raspberry Pi.

Una vez introducido la contraseña mediante el programa Xming obtenemos el escritorio del Raspberry Pi:

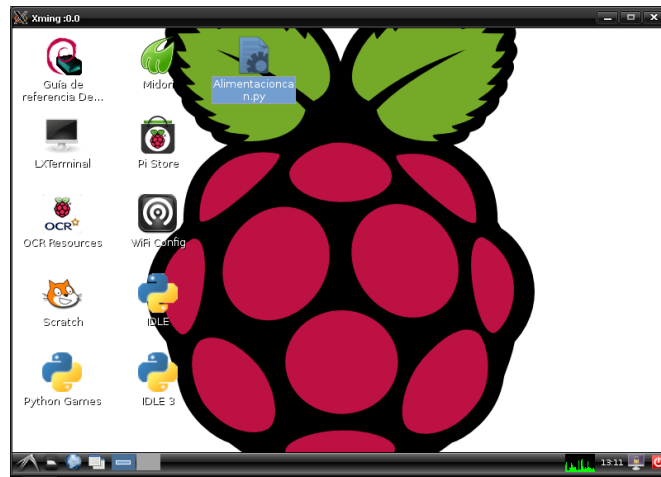


Figura 2.43: Escritorio Remoto del Raspberry Pi corrido desde Xming.

Al obtener el escritorio remoto del raspberry pi, nos dirigimos a un explorador de internet en nuestro caso *chromium* que se encuentra en la barra de inicio del escritorio del raspberry pi.

Inicializado el explorador chromium procedemos a introducir la dirección de las cámaras web que es : 192.168.1.109/camarasweb.html y obtendremos las dos cámaras web en tiempo real.

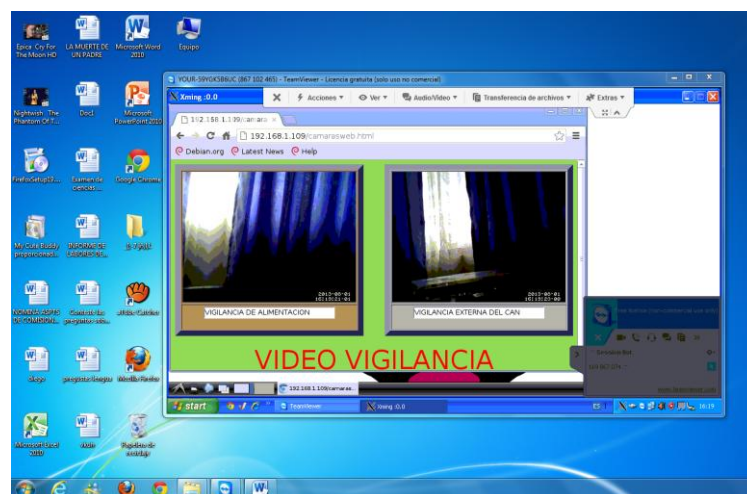


Figura 2.44: Visualización de las cámaras por Internet por medio de Acceso Remoto.

En el siguiente paso observamos en el escritorio del raspberry Pi el programa ejecutable Alimentacioncan.py para abrir y correr el programa que controlara la alimentacion de los animales.

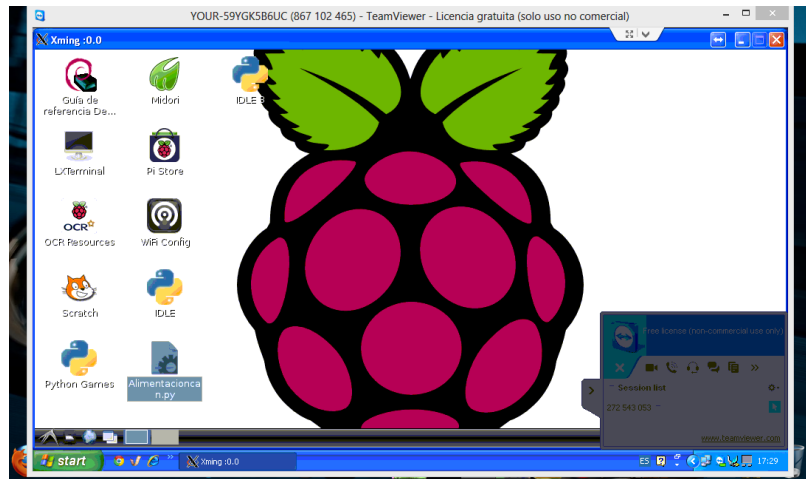


Figura 2.45: Escritorio del Raspberry donde correremos el Programa Alimentacióncan.py.

Una vez abierto realizado doble clic en el programa de alimentacióncan.py nos indica una ventana donde escogemos la opción EJECUTABLE y damos clic.



Figura 2.46: Ventana donde se escoge la opción Ejecutar.

Cuando el programa de Python este corriendo se aparecerá una nueva ventana con los botones que controlan el proceso de alimentación.



Figura 2.47: Ventana del proceso de Alimentación, observado desde el Internet por Acceso Remoto con TeamViewer.

En la siguiente figura observamos una ventana con las camaras web y la ventana con los botones del proceso de alimentacion, se debe destacar que cada ventana es individual tanto como el de las camaras como la ventana del proceso de alimentacion.

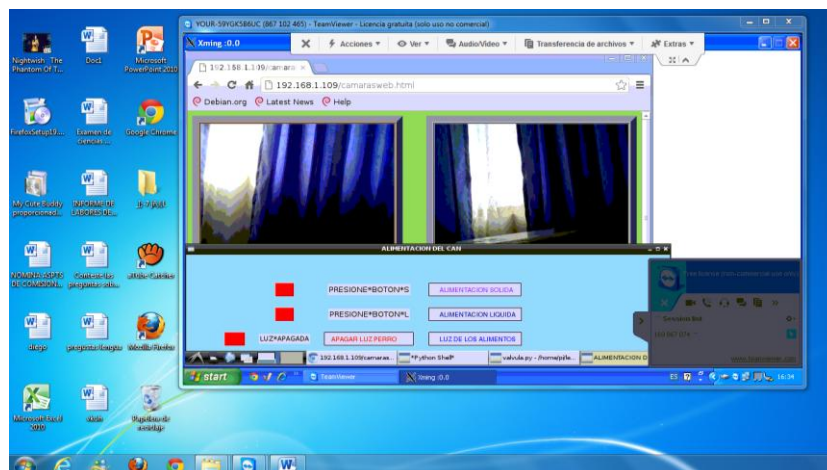


Figura 2.48: Ventanas del proceso de Alimentación y Vigilancia, observado desde el Internet por Acceso Remoto con TeamViewer.

2.6 Integración total del sistema de control, monitoreo y vigilancia.

2.6.1 Diagrama de Bloques del sistema

La figura 2.47 presenta el diseño del circuito de Alimentación, el mismo que contiene:

- Raspberry Pi
- Arduino Uno
- 2 Cámaras Web
- Un Hub Alimentado
- Un Módulo de 4 relés
- Galgas Extensiométricas
- Amplificador operacional LM325N
- Amplificador de Audio
- Lámparas de luminosidad (focos)
- Motor
- Electroválvula.
- Reproductor de música WTV020-SD-16P

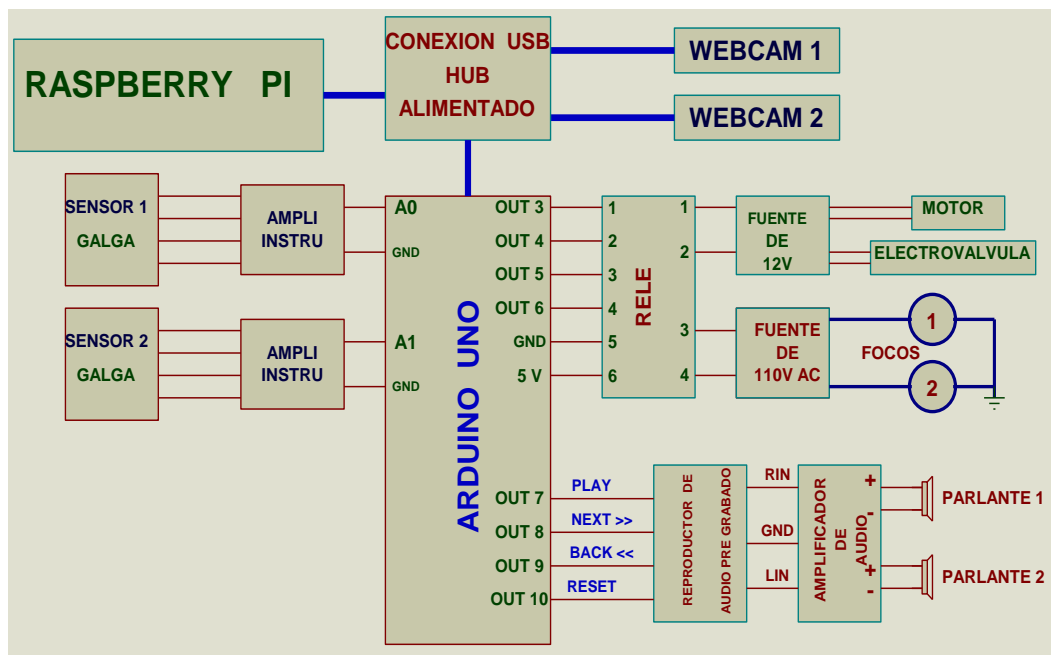


Figura 2.49: Diagrama en bloques de los elementos del sistema.

2.6.2 Diagrama total del circuito de control.

En la figura 2.48 se presenta el diagrama total del circuito del control

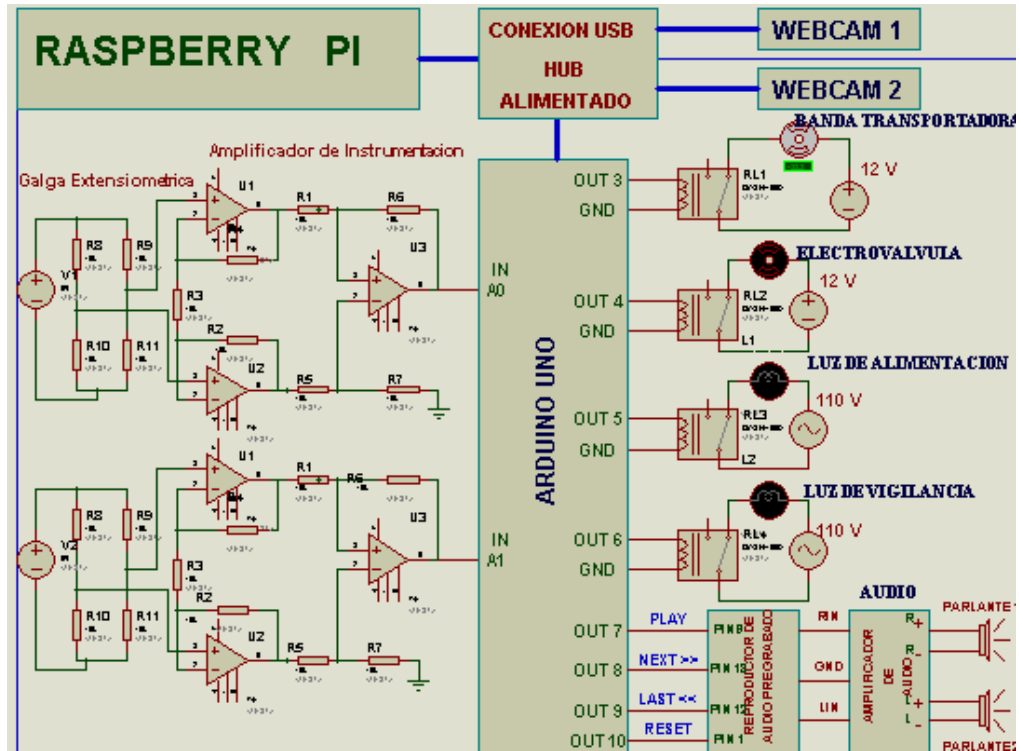


Figura 2.50 Diagrama Total del circuito de control.



Figura 2.51 Conexión total del sistema eléctrico de la alimentación.



a) Cámara de exteriores número 2.



b) Foco de exteriores

Figura 2.52 En las figuras a y b se indican la cámara y una bombilla de exteriores para la vigilancia del can.



Figura 2.53 Sistema de alimentación para canes totalmente terminada.

CAPÍTULO 3

PRUEBAS Y ANÁLISIS DE RESULTADOS

3.1 Descripción física del diseño.

En este apartado se describe las características físicas del proyecto. El cual consta de:

- Una mini computadora llamada Raspberry Pi.
- Una placa Arduino Uno compatible con Raspberry Pi.
- Placa de potencia.
- Placa de amplificador de Instrumentación para los sensores de peso.
- Sensores de: peso.
- Dos cámaras de Vigilancia.
- Un amplificador de audio con dos parlantes.
- Placa para fuente de voltaje de las cargas (Electroválvula, motor).
- Banda Transportadora para el alimento sólido del animal.
- Recipiente de 25 litros que contiene alimentación Líquida para el animal.
- Recipiente de metal Tipo cono que contiene el alimento sólido para el animal.
- Una UPC como energía alternativa para el funcionamiento del sistema.

Todo esto está montado sobre una estructura de metal de 100 x 70 cm. Como se mostró en las figuras del capítulo II.

3.2 PRUEBAS EXPERIMENTALES

Para el desarrollo del sistema se realizaron diferentes pruebas en las cuales se observó algunos inconvenientes con el sistema, estas observaciones se utilizaron para ir mejorando y optimizar al mismo hasta llegar al nivel óptimo requerido para los usuarios. Primero se experimenta con el módulo I/O de la placa Raspberry pi y el software de control desarrollado en Python idle, obteniendo resultados no deseados en las entradas, al verificar que el Raspberry pi no es muy sólido en obtener entradas de tipo analógico, confirmando que su fuerte es en salidas y entradas digitales (0/1).

Se obtiene los resultados esperados de dicha prueba, con algunas observaciones:

- Para el ingreso de señales analógicas se recomendaba utilizar sensores específicos y no más de dos sensores, la cual por su precio y escasez no era muy recomendable utilizar.
- Luego de haber corregido los inconvenientes presentados se hizo la implementación de una placa llamada Arduino que facilito la conexión de sensores analógicos como las galgas extensiométricas que son muy económicas y fácil de conseguir.
- Se debe tomar en cuenta que los sensores dan valores de voltajes muy pequeños para la cual se ha implementado un amplificador de instrumentación para obtener valores deseados e ingresar a la placa Arduino.
- Antes de poner el sistema a funcionar se debe calibrar los sensores de peso, para no desperdiciar el alimento o posibles desbordes del líquido que traerán consecuencias al sistema.

3.3 Pruebas del sistema HTML.

- Luego de haber corregido los inconvenientes presentados se hizo la prueba completa del sistema enlazándose con el Internet, obteniéndose las siguientes observaciones:
- Por ser un control en tiempo real la computadora debe tener un enlace permanente con el Internet mediante el software TeamViewer que es el medio de acceso del usuario con el sistema de alimentación.
- Cuando el proceso esta enlazado al Internet y podemos suministrar el alimento al animal, nos damos cuenta que cuando presionamos el botón de alimentación sólida este empezara a suministrar el alimento y debemos esperar que el proceso termine por ser señal analógica, para continuar con la alimentación de líquido y demás procesos.
- El programa de control en el Internet no debe ser muy pesado ya que esto produce una operación lenta en el sistema de control y vigilancia, por lo cual se recomienda primero abrir la dirección de las cámaras y luego correr el programa que contiene la ventana principal con los

botones que controlan a los actuadores y así empiece el proceso de alimentación.

3.4 Análisis de resultados.

El análisis de pruebas experimentales, permite conocer y determinar las limitaciones y alcances del cumplimiento de objetivos del desarrollo del proyecto implementado.

Tabla 3.1 Ración para la alimentación Diaria de un perro.

| RACION DIARIA (TAZAS / GRAMOS) | | |
|---------------------------------------|---------------|-----------------|
| PESO | GRAMOS | CANTIDAD |
| 1 – 3 Kg. | 50 – 90 g | ½ a 1 taza |
| 3 – 5 Kg. | 90 – 120 g | 1 a 1 ¼ taza |
| 5 – 10 Kg. | 120 – 190 g | 1 ¼ a 1 ¾ taza |
| 10 – 15 Kg. | 190 – 260 g | 1 ¾ a 2 ½ taza |
| 15 – 20 Kg. | 260 – 310 g | 2 ½ a 3 taza |
| 20 – 30 Kg. | 310 – 410 g | 3 a 4 taza |
| 30 – 40 Kg. | 410 – 500 g | 4 a 4 ¾ taza |
| 40 – 50 Kg. | 500 – 590 g | 4 ¾ a 5 ½ taza |
| 50 – 60 Kg. | 590 – 660 g | 5 ½ a 6 ¼ taza |
| 60 – 70 Kg. | 660 – 740 g | 6 ¼ a 7 taza |
| 70 – 80 Kg. | 740 – 800 g | 7 a 7 ½ taza |

- **Cantidad de comidas diarias**

- Hasta los 3 meses: 4 comidas
- De los 3 meses a los 6 meses: 3 comidas
- De los 6 meses a los 17 meses: 2 comidas
- Después de los 18 meses se podrá optar por 1 comida (se recomienda siempre 2 ingestas para una mejor asimilación)

- **Cantidad de Agua diaria para el perro.**

- La cantidad de agua que requiere cada perro depende del tamaño y el peso del animal. Una sencilla regla que puede utilizarse para calcular el líquido

que debe consumir el can consiste en multiplicar por 2,5 el peso de la comida seca que ingiere. De esta forma, un perro que necesita consumir un kilo de pienso diario (croquetas) precisará beber unos dos litros y medio de agua, mientras que un animal que consume dos kilos de alimento seco requerirá unos cinco litros.

- En este subtema se realizan las pruebas de funcionamiento de cada sensor del proyecto, con sus respectivos actuadores.

En la tabla 3.2 y 3.4. Se indican los resultados de las pruebas que se realizaron a la electroválvula para el llenado del bebedero del animal.

Tabla 3.2 Pruebas en la electroválvula.

| ACTUADOR | CONDICIONES | ON (12V) | OFF (0V) |
|-----------------------|--------------------|-----------------|-----------------|
| ELECTROVALVULA | BOTON APAGADO | | √ |
| | BOTON ENCENDIDO | √ | |

Tabla 3.3 Valores de peso del Bebedero del Animal.

| Alimentación Líquida | |
|-------------------------------|-----------------------------------------|
| Valor en peso (Litros) | Valor en mili Voltios del sensor |
| 0 | 236mV |
| 0.5 | 362mV |

En la tabla 3.4 y 3.5. Se indican los resultados de las pruebas que se realizaron al motor de la banda transportadora para el llenado de la alimentación sólida.

Tabla 3.4 Pruebas en la Banda Transportadora

| MOTOR | CONDICIONES | ON (12V) | OFF (0 V) |
|--------------------------------|--------------------|-----------------|------------------|
| BANDA TRANSPORTADOR | BOTON APAGADO | | √ |
| | BOTON ENCENDIDO | √ | |

Tabla 3.5 Valores de peso de la Alimentación sólida del Animal.

| Alimentación Sólida | |
|-------------------------------|-----------------------------------------|
| Valor en peso (gramos) | Valor en mili Voltios del sensor |
| 0 | 63.5mV |
| 226 | 74mV |

3.4.1 Prueba en el sistema de registro de consume de los alimentos.

En la figura 3.1 se presenta un ejemplo realizado en el sistema de registro de consumo de la alimentación Sólida y líquida, la cual se verifica que se guardan los datos ingresados del sensor; es decir es una pantalla de consultas.

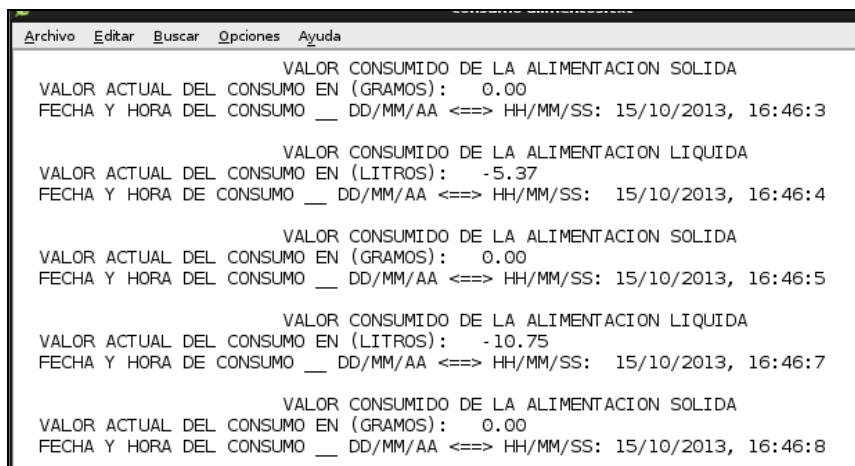


Figura 3.1 Verificación de datos guardados de la Alimentación Sólida y Líquida.

En la figura 3.2 se presenta un ejemplo realizado en el sistema de registro de la alimentación Sólida y Líquida, la cual se verifica que se guardan los datos ingresados cuando se presiona el botón que realiza la alimentación sólida y líquida; es decir es una pantalla de consultas.



Figura 3.2 Verificación de datos guardados al presionar el botón de la Alimentación Sólida y Líquida.

3.5 Análisis Técnico - Económico

En el mercado existe una infinidad de elementos sensores y actuadores para realizar sistemas automáticos con los que se puede trabajar en este sistema, desde los simples pero muy útiles sensores de desplazamiento hasta complejas subestaciones de control. Para este sistema se opta por sensores simples que son de fácil adquisición y bajo costo. Además, como actuadores se utiliza únicamente relés electromagnéticos encargados de controlar el suministro eléctrico de los actuadores como motores y electroválvulas y sistema de iluminación. Todos con su respectivo acondicionamiento de señales.

Como se pretende hacer un control remoto, la mejor herramienta es el Internet, que es un medio de comunicación mundial y accesible en cualquier lugar.

Finalmente se presenta una tabla con los costos correspondientes a la elaboración del proyecto, el cual puede incrementarse en cuanto se requiera cubrir un mayor número de expectativas, comodidades o a su vez colocar otro tipo de sensores y actuadores.

Tabla 3. 4 Valor de adquisición de los componentes localizados en el Diseño.

| COSTOS DEL DESARROLLO DEL SISTEMA DE ALIMENTACION | | |
|----------------------------------------------------------|----------|----------------|
| COMPONENTES | CANTIDAD | VALOR |
| Raspberry Pi | 1 | \$ 80 |
| Arduino Uno | 1 | \$ 28 |
| Cámaras Vx800 | 2 | \$ 57 |
| Placa de 4 relés Arduino | 1 | \$ 28 |
| Tarjetas de memoria SD de 4 Gigas | 2 | \$ 24 |
| Electroválvula | 1 | \$ 15 |
| Motor | 1 | \$ 25 |
| Sensores (Galgas extensiométricos) | 2 | \$ 10 |
| UPC | 1 | \$ 60 |
| Transformador | 2 | \$ 10 |
| Placas | 2 | \$ 5 |
| Lámparas | 2 | \$ 5 |
| Cables de instalación | 5 | \$ 10 |
| Estructura Metálica | 1 | \$ 200 |
| Banda transportadora | 1 | \$ 20 |
| Internet | | \$ 67 |
| Amplificador de Audio | 1 | \$ 26 |
| Varios | | \$ 30 |
| Costos de la Investigación | | |
| Internet | | \$ 300 |
| Movilización | | \$ 100 |
| Costos Administrativos | | |
| Impresiones, documentos, | | \$ 800 |
| TOTAL | | \$ 1900 |

3.6 ALCANCES Y LIMITACIONES.

3.6.1 Alcances

- El prototipo tiene una estructura portátil y segura.
- Brinda la alimentación sin causarles daño alguno.
- Puede suministrar la alimentación en la noche con solo accionar los focos de los recipientes como de los exteriores.
- Funciona de forma remota controlada desde el Raspberry Pi sin conexión a Internet y de forma que se pueda controlar desde el Internet con software gratis.
- El usuario puede vigilar a su mascota cuando está alimentándose o cuando está en los exteriores con las respectivas cámaras.
- El usuario puede utilizar la opción de audio para llamar a su mascota para brindarle su alimentación.
- Se puede usar como método didáctico de enseñanza y prestaciones del Raspberry Pi con Arduino uno.
- Con el Raspberry Pi y el Arduino uno se puede realizar muchas automatizaciones de uso doméstico a precios muy económicos.

3.6.2 Limitaciones

- Realiza un control en las electroválvulas para suministrar el alimento según el valor que se haya puesto a los sensores con referente a su peso en la comida, si se desea suministrar más alimento al animal se tiene que cambiar los valores en su programación.
- La galga extensiométrica soporta un peso de hasta 5Kg de peso.
- Se puede colocar un recipiente para la alimentación de un diámetro inferior a 25cm, y mayores a 15 cm de alto.
- Una vez inicializado el proceso de alimentación no se puede detener hasta que el sensor de la orden de paro al sistema y este detenga a los actuadores.
- No se puede exceder en la instalación de programas en la mini computadora Raspberry Pi.

- No se debe exceder en su corriente al Raspberry Pi.
- No se debe utilizar más de una cámara conectada al Raspberry Pi, si esto sucede se debe adicionar un HUB alimentado para las conexiones de las cámaras.

CAPITULO 4

CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

1. Con el desarrollo del proyecto se logró cumplir el objetivo principal propuesto, que consiste en diseñar y construir un sistema de control, monitoreo y vigilancia por cámara web vía internet para la alimentación de animales domésticos, utilizando software libre, dando una aplicación adicional al servicio de Internet que disponemos en este caso conocer el estado de su mascota cuando tenga que abandonar su residencia.
2. Mediante la implementación de las cámaras web hemos cumplido con el objetivo de vigilar a nuestra mascota, cerciorarnos si se alimentó adecuadamente y observar a nuestra mascota en vivo y en directo desde cualquier lugar en el cual nos encontremos.
3. Se obtuvo un sistema de bajo costo mediante el esfuerzo personal ya que se utiliza una mini computadora denominada Raspberry Pi que comparada con una computadora normal es más económica. Además como todo sistema de control, monitoreo y vigilancia este permite obtener seguridad y tranquilidad para los miembros de su familia al poder observar y controlar la alimentación de su mascota, ya que desde cualquier lugar se puede conocer el estado de las variables controladas: control de la alimentación sólida y líquida, además la vigilancia por medio de cámaras ubicadas estratégicamente.
4. La utilización del software libre permitió reducir los costos del sistema construido, de tal manera que es accesible para la mayoría de usuarios.
5. Se comprueba que la comunicación entre el Hardware y Software del Raspberry Pi y el Arduino Uno a través del puerto USB, es válida y eficaz porque se ha logrado el encendido y apagado de lámparas para

iluminación y los actuadores que permiten controlar la cantidad de alimento para los canes.

6. Se ha mejorado la seguridad de los Animales Domésticos (Perros) que son para muchos un bien apreciado, porque se ha implementado un módulo de control y vigilancia de acceso a través de Internet por medio de software libre o llamado TeamViewer que nos facilita el Acceso Remoto a una computadora que estará en la misma red con el Raspberry Pi y desde ahí poder controlar y vigilar la alimentación del can.
7. Se logró conseguir por medio de una programación adecuada un tiempo promedio de retardo de encendido y apagado de los periféricos de salida de 4 segundos, tiempo suficiente para el correcto funcionamiento del sistema de control, monitoreo y vigilancia de la mascota.
8. Se determinó la facilidad de conexión de los periféricos de salida de la tarjeta Arduino Uno con los actuadores, para el control de lámparas, electroválvulas, y motor utilizados en el proceso de alimentación.
9. Gracias a los avances tecnológicos y el conocimiento adquirido se puede desarrollar sistemas complejos de monitoreo, control y vigilancia dependiendo de las necesidades que el usuario requiera.
10. Para el procesamiento de los datos adquiridos se utilizó la tarjeta Arduino Uno, debido a que en este sistema se puede trabajar con la velocidad de comunicación del puerto, trabajar con señales analógicas y digitales y es mucho más económica que otras tarjetas que se utilizarían para el mismo fin.
11. Ciertos componentes requieren que estén continuamente energizados por lo cual se utiliza una UPS, que se encarga de entregar energía en caso de que exista corte del suministro público.
12. Con el sistema de registros se tiene una mejor verificación de todos los datos de consumos de la alimentación sólida y líquida que se realizan y se puede verificar cuanto de alimentación queda en los recipientes.

13. Se utilizó lámparas en la parte de los recipientes de alimentación como en lo alto de la estructura debido a que el usuario podría dejar para la noche la alimentación de su mascota.
14. Se utilizó un sistema de reproducción de sonido donde previamente está grabado la voz del usuario con diferentes archivos de voz, para que la mascota este asociado con él durante el proceso de alimentación.
15. La operación del sistema está diseñado para cubrir un máximo de 5 días de alimentación, en ausencia del propietario de la mascota por la capacidad de los recipientes de alimentación.
16. El Raspberry Pi no es un ordenador que pueda remplazar a ningún otro. No está diseñado para ser una PC corriente y tampoco funciona de forma comparable. Este pequeño dispositivo tampoco hará que te desprendas de tu PC o reproductor multimedia ya que no está concebido con ese objetivo.
17. El Raspberry Pi es ideal para aprender programación, hardware, software y tecnología en general, con el apoyo de una gran comunidad de usuarios que no deja de crecer. Su precio es absolutamente justificable. Sólo hay que tener muy claras las limitaciones del aparato, y que después de adquirirlo tenemos que pasar varias horas con él antes de que puedas aprovechar su potencial.

4.2 RECOMENDACIONES

Además vemos oportuno sugerir las siguientes recomendaciones que podrán ser tomadas en cuenta para futuros proyectos:

1. Si se desea mejorar la velocidad de procesamiento del Raspberry Pi, se recomienda no instalar programas que no se vaya a utilizar. Si se toma en cuenta estas recomendaciones para futuros proyectos similares, se lograría que el tiempo de encendido y respuesta de los periféricos y las cámaras se reduzcan.
2. Como el sistema está continuamente energizado se recomienda que exista suficiente ventilación para los equipos.

3. Se recomienda hacer más convenios con empresas nacionales e internacionales para que se puedan desarrollar proyectos prácticos con tecnologías y equipos actuales.
4. Incentivar a los estudiantes a foros tecnológicos, actividades sociales y a fomentar el compañerismo con actividades grupales, ya que esos valores se van perdiendo con el pasar del tiempo.
5. Para mejorar la investigación y obtener mejores resultados se recomienda mejorar el servicio de internet, entre ellos aumentar el ancho de banda, ampliar la cobertura de internet inalámbrica en la Universidad de las Fuerzas Armadas - ESPE Extensión Latacunga.

BIBLIOGRAFÍA

• REFERENCIAS EN LIBROS

- Sensores y Acondicionadores de señal
Pallo Areny, editorial Alfaomega, Tercera edición,
México 2001.....Pág. 69,70
- Internet.
Harley Hahn, editorial Osborne Mcgran Hill, Segunda edición,
España 1997.....Pág.33
- Word Wide Web
Gómez Aynau A, editorial Osborne Mcgran Hill,
España 1996.....Pág. 34

NETGRAFÍA

- https://www.waypoint.cl/layout.php?a=servicio_flotagps
- <http://www-ei.eui.upm.es/Asignaturas/BD/BD/docbd/tema/Arquitectura.pdf>
- <http://www.slideshare.net/jpbthames/arquitectura-de-sistemas-distribuidos>
- <http://www.raulcarretero.com/2012/04/17/por-que-y-cuando-elegir-un-sistema-domotico-centralizado-o-distribuido/>
- http://www.axis.com/es/products/video/about_networkvideo/evolution.htm
- <http://es.wikipedia.org/wiki/Actuador>
- <http://www.monografias.com/trabajos15/actuadores/actuadores.shtml>
- http://educativa.catedu.es/44700165/aula/archivos/repositorio/4750/4928/html/3_clasificacin_de_transductores.html
- <http://www.slideshare.net/veronik211/clasificacin-de-los-transductores>
- <http://www.tusequipos.com/2009/10/19/microsoft-lifecam-vx-800-una-camara-web-simple-y-barata/>
- <http://www.monografias.com/trabajos37/medios-transmision/medios-transmision2.shtml>

- <http://www.monografias.com/trabajos37/medios-transmision/medios-transmision.shtml>
- <http://www.raspberrypi.org/phpBB3/viewtopic.php?t=4751>
- http://es.wikipedia.org/wiki/Raspberry_Pi
- <http://arduino.cc/en/Main/arduinoBoardUno>
- <http://www.bricogeek.com/shop/arduino/305-arduino-uno.html>
- <http://www.cortoc.com/p/arduino.html>
- <https://sites.google.com/site/temasdedisenoymanufactura/arduino>
- <http://es.scribd.com/doc/35568727/Puertos-y-Conectores-de-La-PC>
- <http://www.monografias.com/trabajos5/datint/datint.shtml>
- http://es.wikipedia.org/wiki/Conexi%C3%B3n_a_Internet
- <http://www.monografias.com/trabajos13/modosi/modosi.shtml>
- http://es.wikipedia.org/wiki/Modelo_OSI
- <http://oscarbasededatos2012.blogspot.com/2012/11/arquitectura-clienteservidor.html>
- http://es.wikipedia.org/wiki/Direcci%C3%B3n_IP
- <http://tecnosaurios.blogspot.com/2011/10/medios-de-acceso-internet.html>
- <http://www.mtxtunnel.com/blog/ejemplo-control-flotas-1-envio-de-posicion-gps-y-es-a-servidor-por-gprs/>
- http://www.broadband.gov/spanish/broadband_types.html

- **Ingeniería y Consultoría Domótica**

Copyright Domótica Viva, S.L.

Disponible en World Wide Web:

- <http://www.domoticaviva.com/presente.htm>

- **Kioskea. Net**

Modos de Transmisión

Disponible en World Wide Web:

- <http://es.kioskea.net/contents/transmission/transmode.php3>

- **Conexión a internet.**

Alegsa, Todos los derechos reservados © 1998 - 2012 - ALEGSA – Santa Fe, Argentina.

Disponible en World Wide Web:

- <http://www.alegsa.com.ar/Notas/135.php>

- **Cámara Web**

Wikipedia

Disponible en World Wide Web:

- http://es.wikipedia.org/wiki/C%C3%A1mara_web

- **Verificación de las direcciones IP**

Advanced IP Scanner

Copyright 1999-2013 Famatech.

Disponible en World Wide Web:

- <http://www.advanced-ip-scanner.com/>

- **Instalación del sistema operativo en Raspberry Pi.**

IEEE UNIVERSIDAD DE OVIEDO STUDENT BRANCH

AÑO 2012

Disponible en World Wide Web:

- <http://ieeesb-uniovi.es/talleres-charlas/linux/raspberry-pi/>

- **RASPBERRY PI descarga del sistema Operativo**

Fundación Raspberry Pi

Actualización el 5 de Mayo del 2013.

Disponible en World Wide Web:

- <http://www.raspberrypi.org/downloads>

- **Software Win32 para cargar imagen en la tarjeta SD del Raspberry Pi.**

Image Writer para Windows

Inscrito 2009-04-19 por Michael Casadevall

Disponible en World Wide Web:

- <https://launchpad.net/win32-image-writer>

- **Software XMING Enlace a Acceso Remoto con el Raspberry Pi**

Sourceforge

Copyright © 2013 Dice.

Disponible en World Wide Web:

- <http://sourceforge.net/projects/xming/postdownload?source=dlp>

- **Tutoriales del Raspberry Pi**

Tecbolivia.com

Disponible en World Wide Web:

- <http://www.tecbolivia.com/index.php/articulos-y-tutoriales-microcontroladores/55-tutorial-de-raspberry-pi>
- <http://sistadap.blogspot.com/>

- **Tutoriales de Programación en Python**

Resumen de interfaz Grafica

Disponible en World Wide Web:

- <http://gmendezm.blogspot.com/2012/12/tutorial-tkinter-python-gui.html>
- <http://pharalax.com/blog/python-desarrollo-de-interfaces-graficas-con-tkinter-labelsbuttonsentrys/>
- http://eliluminado.alwaysdata.net/Guia_Tkinter
- <http://nuestrapizarra.allalla.com/2011/08/introduccion-a-tkinter/>

- **Instalación de las cámaras en Raspberry Pi**

2013 DIVERTEKA

Disponible en World Wide Web:

- <http://www.diverteka.com/?p=709>
- <http://www.diverteka.com/?p=673>

- **Ejemplos de Conexión del Raspberry Pi con Arduino Uno**

Arduino Uno + Raspberry Pi + PHP + LED (update 02/06/2013)

Disponible en World Wide Web:

- <http://forum.arduino.cc/index.php?PHPSESSID=s10gvlq1ro39qvpantup123443&topic=168306.msg1252715#msg1252715>

- **Decodificador de audio Arduino.**

Miércoles 20 de febrero 2013

Arduino de alro

Disponible en World Wide Web:

- <http://arubia45.blogspot.com/2013/02/decodificador-wtv020-sd-16p-arduino.html>

- **Tipos de Conexión a Internet**

INTEF Instituto Nacional de Tecnologías y de Formación de Profesorado.
España- Madrid.

Disponible en World Wide Web:

- <http://recursostic.educacion.es/usuarios/web/ayudas/54-conexiones-a-internet-bis>

- **Direccionamiento de IP**

Disponible en World Wide Web:

- <http://www.euskalnet.net/amesperuza/ip.html>

ANEXOS

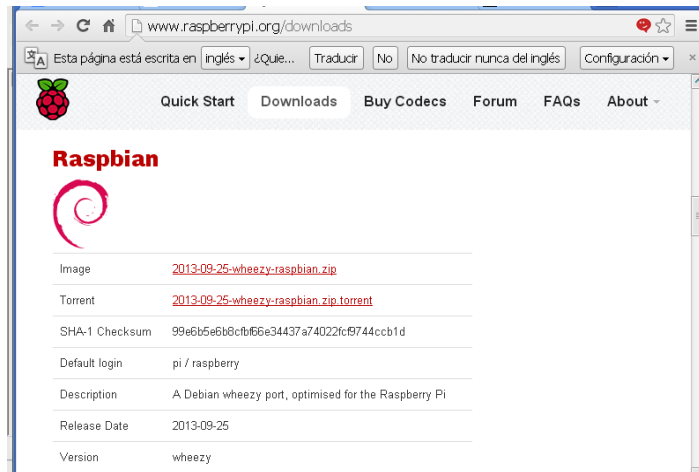
ANEXO A

Instalación y Configuraciones de la mini computadora

RASPBERRY PI

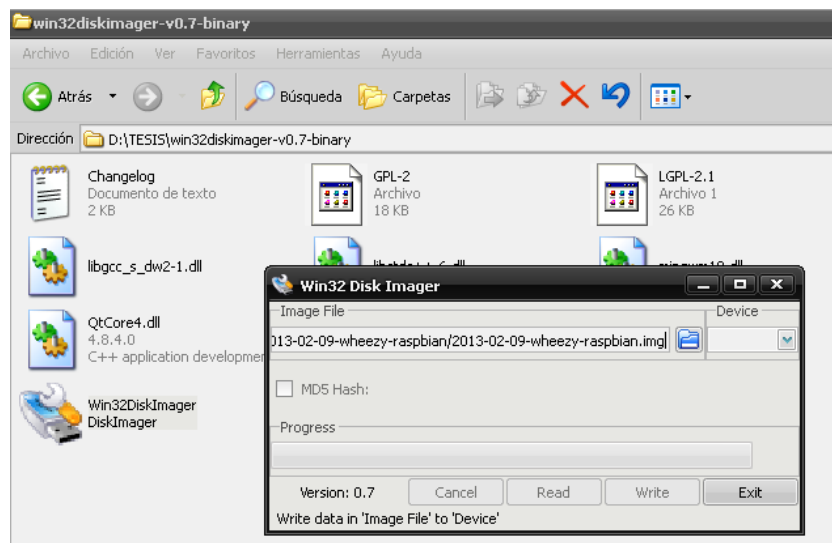
Instalación y Configuraciones de la mini computadora RASPBERRY PI.

Antes de encender por primera vez habrá que instalar una distribución preparada y copiarla en la tarjeta SD. En la web oficial hay unas cuantas distribuciones listas para utilizar. Usaremos Debían por ser la más versátil lo encontraremos en <http://www.raspberrypi.org/downloads>.



Anexo A.1 Ventana de descarga del software de Raspberry pi.

Para copiar el archivo ZIP en la tarjeta SD nos bajaremos el Win32diskimager y si estamos en Linux instalamos de los repositorios el ImageWriter en <https://launchpad.net/win32-image-writer/+download>. Ambos programas permiten elegir el fichero bajado y copiarlo en la tarjeta SD.



Anexo A.2 Programa que carga el sistema operativo en la tarjeta SD

Encendido por primera vez

Para empezar necesitaremos lo siguiente (Información extendida):

- Raspberry Pi
- Cargador de móvil con micro USB (5V y > 700mA).
- Cable HDMI conectado a una pantalla.
- Ratón y teclado USB conectados.
- Tarjeta SD (mínimo 2GB).
- Cable de red (con IP dinámicas activadas).

Una vez que lo hayamos configurado todo, lo único que necesitará es la alimentación de 700mA, todo lo demás dependerá para el uso que le demos ¿controlará un robot, servidor web (cable de red), reproductor multimedia (teclado+HDMI)?

Para configurar la pantalla y algún detalle más se hace con el fichero “config.txt” de la carpeta /boot/ que es accesible desde cualquier ordenador (por defecto el fichero no existe), un poco más adelante profundizaremos en este asunto.

Primeros comandos de Linux

Antes de seguir vamos a definir algún concepto básico:

- sudo => Al poner esto al principio de un comando estaremos ejecutando el mismo con privilegios de administrador, ¡cuidado cuando lo ejecutemos!
- # => Carácter de escape para los comentarios (igual que el // de C)
- ssh pi@192.168.2.107 => Conexión remota al ordenador en la IP 192... con el usuario “pi”, una vez establecida la conexión (pedirá usuario y contraseña) se podrá ejecutar los comandos de la consola de Linux desde el otro ordenador como si estuviéramos usando el teclado del Raspberry Pi (es algo común a todos los Linux). En Windows el programa más usado es el Putty (web oficial, descarga directa para Windows).
- Para saber más podemos buscar “comandos Linux” (ejemplo, otro más completo). Algunos de los más útiles son: mv (*mover*), rm (*borrar*), ls (*listar contenido carpeta*), y nano (*editor de textos*).

Ahora supongamos que ya hemos instalado la SD y encendido el Raspberry Pi. Tras unos minutos que tarda en configurarse nos pedirá el usuario/contraseña (en Debian será pi/Raspberry). Comencemos a hacer cosas:

Activar el SSH, para poder conectarnos desde otro ordenador:

```
sudo service ssh start
```

```
sudo insserv ssh
```

Cambiar la contraseña, que por defecto será “*raspberrypi*“:

```
sudo passwd pi
```

Apagar desde la consola (aunque al final tendremos que desconectar la alimentación):

```
sudo shutdown -h now
```

Espacio disponible en la tarjeta SD:

```
df -h
```

Configuración del hardware

Editar el fichero de inicio (ya hemos visto antes para qué servía, y el sitio con la documentación):

```
sudo nano /boot/config.txt
```

De lo más importante (MUY recomendable visitar la explicación extendida) a escribir en el fichero será:

```
sdtv_mode=2 # Salida analógica en formato PAL (el europeo)
```

```
hdmi_group=2 # Tipo de salida HDMI (1 por defecto): 1=CEA y 2=DMT
```

```
hdmi_mode=16 # Resolución de salida (depende del anterior)
```

Para saber qué formatos admite nuestra pantalla (es decir, los valores que podemos poner en *hdmi_mode*) ejecutaremos los siguientes comandos en la consola:

```
/opt/vc/bin/tvservice -m CEA
```

```
/opt/vc/bin/tvservice -m DMT
```

Y para saber cuál es el modo que tenemos actualmente activado:

```
/opt/vc/bin/tvservice -s
```


Entorno gráfico

Para usar un entorno gráfico (es decir, usar el ratón) ejecutaremos este sencillo comando:

```
startx
```

Si queremos que se inicie automáticamente podremos hacer lo siguiente:

```
sudo nano /etc/profile
```

Y añadir al final del archivo añadir una línea nueva que ponga “startx” (sin las comillas).

Si además queremos que no pida el usuario/contraseña al iniciarse podemos seguir los pasos de esta web y veremos a continuación:

```
sudo nano /etc/inittab
```

Buscar el siguiente texto y añadir un # delante de ella:

```
1:2345:respawn:/sbin/getty 115200 tty1
```

Para finalizar, escribiremos lo siguiente debajo de la línea que acabamos de comentar:

```
1:2345:respawn:/bin/login -f pi tty1 </dev/tty1 >/dev/tty1 2>&1
```

Pulsamos *Control+X* (para salir) y confirmaremos que sí queremos guardar los cambios.

Instalación de programas

La forma de proceder siempre será actualizar los paquetes, instalar las actualizaciones y luego instalar. Podemos instalar directamente si durante ese día ya hemos comprobado las actualizaciones:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

(si existen actualizaciones diremos que sí queremos ejecutarlas).

Y el siguiente comando es para instalar un programa:

```
sudo apt-get install XXXXX
```

Siendo XXXXX el nombre del programa a instalar.

NOTA: Si tenemos una tarjeta SD de menos de 3GB nos convendrá ejecutar “sudo apt-get clean” cada pocas instalaciones, pues nos ahorraremos unos 150-

200 de megas. Si por el contrario, tenemos más de 1GB libre en la tarjeta no es necesario ejecutarlo nunca.

Españolizando el Raspberry Pi

Por defecto todo viene en inglés y los símbolos también estarán cambiados. Por ejemplo, el guión “-” se encontrará justo a la derecha del 0 (y el = a continuación), el # se encuentra en el número 3 ¡y nuestra querida ñe no está! Para solucionarlo ejecutaremos los siguientes pasos:

```
sudo dpkg-reconfigure keyboard-configuration
```

Yo elegí el que marca por defecto (genérico de 105 teclas), pero si conocemos el modelo de nuestro teclado podemos buscarlo en la lista, una vez elegido pulsamos ENTER para continuar. El “keyboard layout” a elegir se encuentra en “other //Spain //Spain“. Respecto al AltGr escogemos “The default for the keyboard layout” y en lo referente al Control+Alt+Retrosceso recomiendo decirle que SÍ queremos asignarle la combinación de teclas (nos será muy útil si se cuelga el sistema gráfico).*ENTER* para finalizar.

```
sudo setupcon
```

Con este sencillo comando conseguimos que el arranque se efectúe rápido (en caso de no ejecutarlo el arranque se ralentiza llegando a tardar hasta 1 minuto extra mientras está en “*Setting up console font and keymap*“). Este comando hay que ejecutarlo siempre que reconfiguremos el teclado.

```
sudo dpkg-reconfigure tzdata
```

Aquí se elige el uso horario, el nuestro será *Europe//Madrid*. Usaremos las flechas para marcar y el *ENTER* para aceptar.

```
sudo dpkg-reconfigure locales
```

Idioma español. Elegimos *es_ES.UTF-8* (pulsando espacio para que aparezca un *), y le quitamos el * al que pone *en_GB*. Con *ENTER* aceptamos para seguir. Ahora elegimos *es_ES* (con las flechas, marcándolo en rojo) y terminamos con *ENTER*.

Servidor Web

Hay que instalar/configurar 6 tipos de programas: Acceso SSH (para administrar remotamente), Apache (servidor), PHP (para las páginas dinámicas), MySQL (base de datos), phpMyAdmin (para administrar fácilmente las bases de datos) y Cliente de DDNS (para mantener actualizada la IP de casa).

Para ello usaré el manual publicado en: [y pondré aquí un resumen](#) (recomiendo visitar la web original, pues viene todo explicado paso a paso).

Habilitar SSH

Nos permitirá ejecutar los comandos de la consola desde otro ordenador y de paso podremos acceder a los ficheros del Raspberry Pi usando un cliente de SFTP (recomiendo utilizar FileZilla).

Comandos a ejecutar (en el Raspberry Pi):

```
sudo service ssh start
```

Para iniciar el servicio ssh.

```
sudo insserv ssh
```

Para que se inicie siempre que encendamos el Raspberry Pi se pueda utilizar el SSH. Desde que ejecutamos esto, los siguientes comandos que hay en este manual se pueden ejecutar desde el propio Raspberry Pi o desde otro ordenador (conectándonos al Raspberry Pi usando SSH).

Servidor Web – Apache

Para que todo funcione se necesita instalar el propio servidor, el más común es Apache. Antes de empezar hay que prepararse para que todo funcione. Primero crearemos el grupo de usuarios para el servidor:

```
sudo groupadd www-data
```

```
sudo usermod -a -G www-data www-data
```

Y después actualizamos los repositorios para evitar fallos en la instalación:

```
sudo apt-get upgrade
```

Ahora ya podremos instalar el Apache:

```
Sudo apt-get install apache2
```

Si todo ha ido bien, podremos visitar desde el navegador la página web de prueba, se titulará “It works!” y un par de frases más. Si no funcionara nos daría un error de no encontrado. Para poder visitar la web desde otro ordenador de nuestra habrá que conocer la IP del Raspberry Pi (en la web de las instrucciones hay capturas de pantalla más esclarecedoras) y para saber la IP se ejecutará el comando:

```
ifconfig
```

En el que el apartado “*inet addr:*” veremos la IP (de la red eth0). Habitualmente será algo del tipo 192.168.10.xx

NOTA: La IP 127.0.0.1 es una IP reservada que se usa para conectarse al propio ordenador o para comprobar que funciona la Red.

Accedemos con el navegador a la IP del Raspberry Pi desde nuestro ordenador:***http://192.168.2.105*** (en tu caso será otra IP diferente). Si queremos añadir más ficheros, carpetas y similar se encontrarán en el directorio ***/var/www/*** del Raspberry Pi. Hagamos un Hola mundo:

```
sudo nano /var/www/holamundo.htm
```

Y escribimos lo siguiente en el fichero:

```
<html>
<head>
<title>Titulo de la web</title>
</head>
<body>
<h1>Hola mundo</h1>
<p>Acabo de escribir mi primera web en mi Raspberry Pi.<br />&iquest;Mola,
no?</p>
</body>
</html>
```

Para cerrarlo, como siempre, Control+X, nos preguntará si quieres guardar el búfer (pulsando la S, para decir que sí) y luego ENTER para guardar el fichero con el nombre que pusimos antes. Ahora si visitamos: ***http://192.168.2.105/holamundo.htm***veremos nuestra página web.

SERVIDOR WEB – PHP

Con Apache se pueden mostrar páginas con HTML, pero para hacer webs “de verdad” será necesario poder ejecutar contenido dinámico, y en esto el PHP es el rey. Procedamos a su instalación:

```
sudo apt-get install php5
```

Tras terminar, instalemos los demás paquetes que podríamos necesitar (en la web original ponen más paquetes):

```
sudo apt-get install libapache2-mod-php5 libapache2-mod-perl2 php5 php5-cli  
php5-common php5-curl php5-dev php5-gd php5-imap php5-ldap php5-mhash  
php5-mysql php5-odbc
```

Tras esta segunda instalación, reiniciar (*sudo reboot*).

Y para comprobar que funciona, haremos la web de pruebas de PHP, que consiste en una única función (son los mismos pasos que para crear la web en HTML):

```
sudo nano /var/www/phpinfo.php
```

Y luego rellenar el fichero con lo siguiente:

```
<?php  
    phpinfo();  
?>
```

Si al entrar en ***http://192.168.2.105/phpinfo.php*** (usar la IP que corresponda) vemos una web con mucha información es que todo ha funcionado correctamente.

ANEXO B

Instalación y Configuraciones de la Tarjeta Electrónica Programable Arduino Uno.

Instalación y Configuraciones de la Tarjeta Electrónica Programable Arduino Uno.

Configuración del Software

Conectamos el Arduino Uno a la raspberry Pi luego monitor, mouse, teclado y al final la alimentación.

Nosotros usamos la imagen Occidentalis

Sudo apt-get upgrade

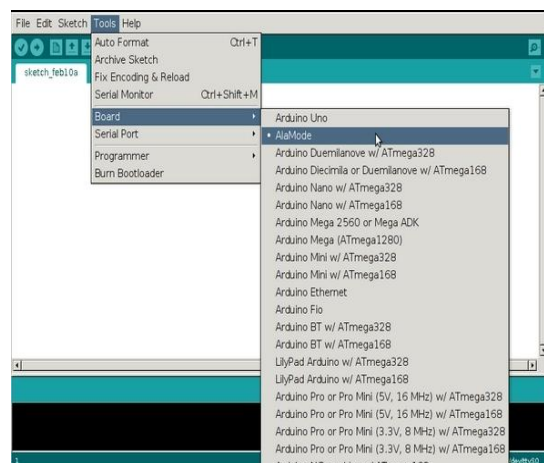
Sudo apt-get install Arduino

Ahora desde la interfaz gráfica se inicia el Arduino IDE como se aprecia en la siguiente figura



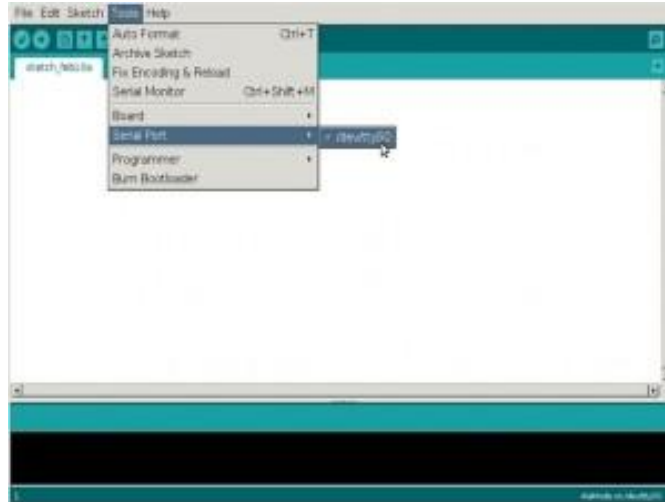
Anexo B.1 Inicio del programa Arduino en Raspberry pi.

Ahora verificamos que esté seleccionado el Arduino Uno, siguiendo el Menú->Tools->Board->Alamode



Anexo B.2 Ventana del programa Arduino

Y se selecciona la puerta desde Menú->Tools->Serial Port-> dev/ACOM0



Anexo B.3 Habilitación del puerto Serial en Arduino.

Ahora estamos en condiciones de programar y usar el Arduino uno.

ANEXO C
Instalación del TeamViewer

Instalación del TeamViewer

Funcionamiento del TeamViewer.

El software no puede usarse instalándolo en el sistema, aunque la versión 'Quick Support' puede ejecutarse sin necesidad de instalación. Para conectarse a otro equipo, ambos deben estar ejecutando TeamViewer. Para su instalación, requiere acceso de administrador, pero una vez instalado puede ser usado por cualquier usuario del ordenador.

Cuando se inicia en un equipo, el programa genera una ID y una contraseña (también permite que el usuario establezca su propia contraseña). Para establecer una conexión entre un equipo local y otro remoto, el usuario del equipo local debe ponerse en contacto con el otro y este debe indicarle la ID y la contraseña. Una vez hecho esto, se introducen en el programa TeamViewer que se está ejecutado en el ordenador local.

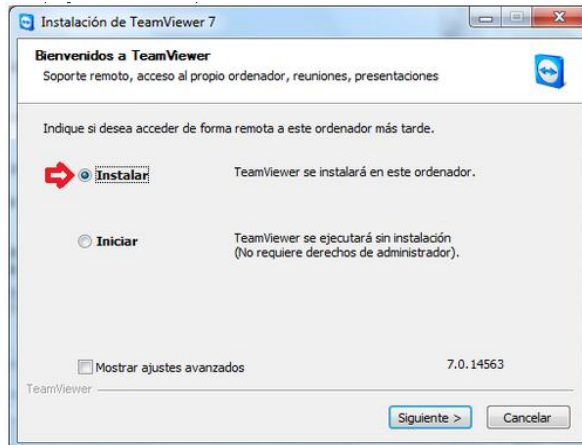
Para comenzar una reunión en línea, el ponente proporciona la ID de dicha reunión a los participantes. Estos se unen a la sesión utilizando la versión completa del programa, o accediendo a la versión para navegador web mediante dicho ID. También es posible programar una reunión con antelación.

Instalación de TeamViewer

Descarga TeamViewer (para Windows, para otros sistemas según en qué tipo de sistema operativo desees manejar según tu necesidad).

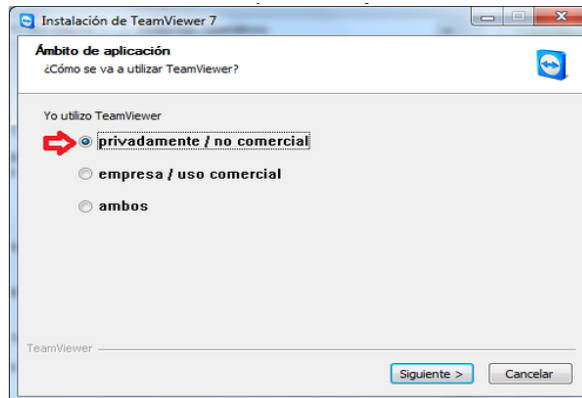
Una vez descargado el archivo de instalación, dale doble clic. En la ventana que aparece haz clic en "Ejecutar"

En la ventana que se abre, marca la casilla "Instalar" y dale clic a "Siguiente". (Marcar "Iniciar" solo si vas usar el programa una sola vez)



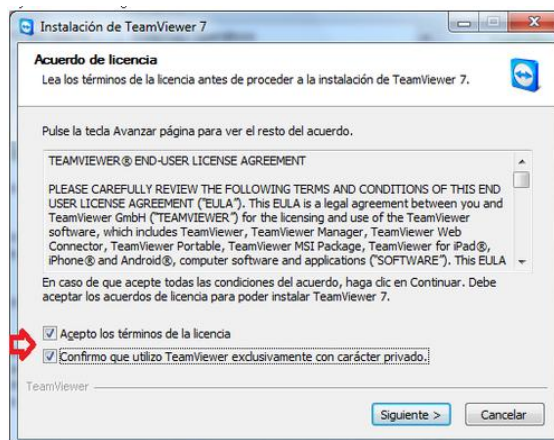
Anexo C.1 Ventana de Instalación de TeamViewer.

Selecciona "Privadamente / no comercial" y dale clic a "Siguiete"



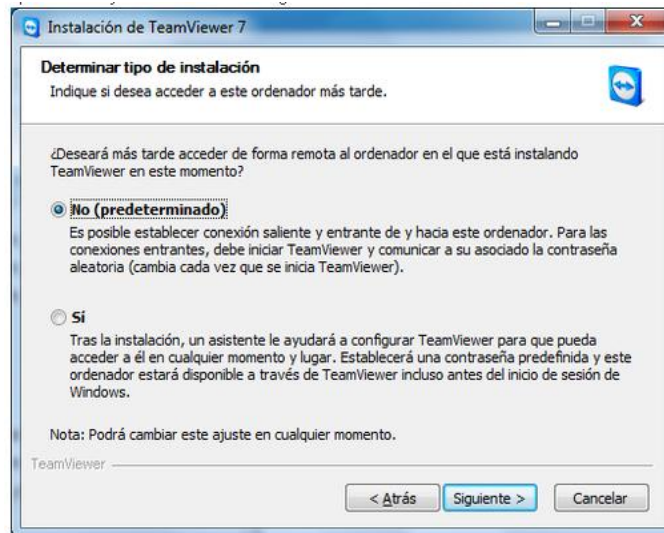
Anexo C.2 Ventana donde se usa el uso comercial o privado de TeamViewer.

Marca las casillas para acepta los términos de licencia y confirmar que usas TeamViewer para uso privado y dale clic a "Siguiete"



Anexo C.3 Ventana donde se afirma el uso privado de TeamViewer.

En la ventana tipo de instalación selecciona una de las opciones. Nosotros dejamos seleccionada la opción por defecto y hacemos clic en "Siguiente"



Anexo C.4 Determinación de Instalación.

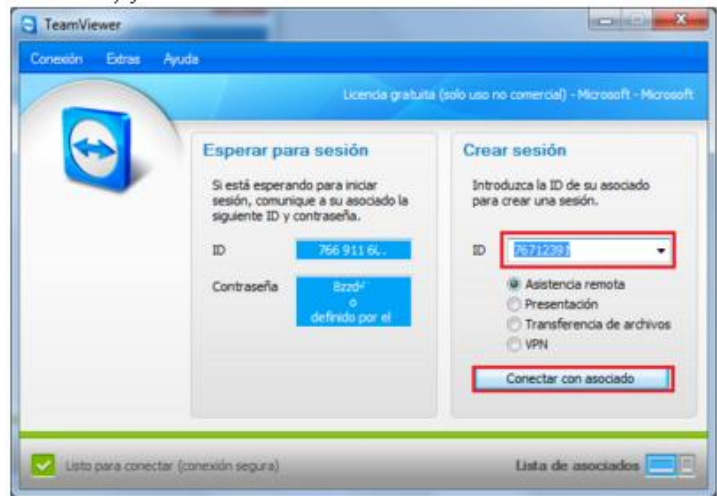
Se instala el programa correctamente y estará listo para poder utilizarlo.

Uso

Una vez terminado el proceso de instalación aparece la ventana principal de TeamViewer. En esta aparece, a la izquierda, el ID y la contraseña que deberá ingresar tu asociado para que se pueda conectar a tu máquina. En cambio, para que puedas iniciar sesión y tomar control de la máquina de tu asociado debes ingresar su ID y contraseña.

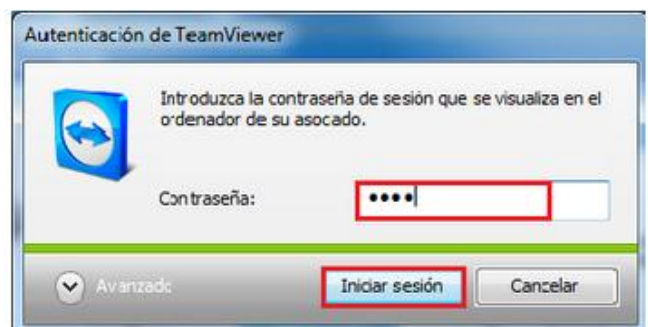
Recuerda que el asociado debe también tener instalado el programa y al término de la instalación o cada vez que ejecute el programa le aparecerá una ventana similar a la que te aparece indicando su ID y contraseña, la que debe comunicarte para que puedas iniciar la conexión.

Ingresa la ID (de tu asociado) y haz clic en "Conectar con asociado"



Anexo C.5 Ventana Abierta de TeamViewer.

En la ventana que aparece ingresa la contraseña (de tu asociado) y haz clic en "Iniciar sesión"



Anexo C.6 Ventana de verificación de la contraseña en TeamViewer.

Y listo Te aparecerá automáticamente el Escritorio de tu asociado. Ahora puedes trabajar en él como si estuvieras enfrente de su PC.



Anexo C.7 Escritorio del asociado activado con TeamViewer.

ANEXO D
PROGRAMACION EN PYTHON

PROGRAMACION EN PYTHON

DESARROLLO DEL PROGRAMA EN PYTHON:

```
#!/usr/bin/python
import RPi.GPIO as GPIO
import serial
import Tkinter as tk
from Tkinter import *
import time
cero = 0.0
try:
    arduino = serial.Serial( '/dev/ttyACM0', 9600 )
except:
    print "Cannot conect to the port"
def LED1():
    arduino.write('U')
    time.sleep(10)
    button17= Button(root, text="  ",bg="green",font=("Helvetica", 10))
    button17.grid(row=4, column=4)
    label39 = Label(root, text="*****FINALIZADO*****", font=("gothica", 12),
bg="sky blue",fg="dark green")
    label39.grid(row=4, column=6)
    arduino.write('X')
    archivo_salida = open ("Registro botón S.txt","a")
    archivo_salida.write("\n\n BOTON PRESIONADO DE LA
ALIMENTACION SÓLIDA \n ")
    archivo_salida.write(" FECHA Y HORA DEL CONSUMO __ DD/MM/AA
<==> HH/MM/SS: ")
    archivo_salida.write (imprimir _ fecha ())
    archivo_salida.close()
def LED2():
    arduino.write('D')
```

```

time.sleep(82)
button18= Button(root, text="  ",bg="green",font=("Helvetica", 10))
button18.grid(row=6, column=4)
label40 = Label(root, text="****FINALIZADO****", font=("gothica",
12),bg="sky blue", fg="dark green")
label40.grid(row=6, column=6)
arduino.write('Y')
archivo_salida = open ("Registro botón L.txt","a")
archivo_salida.write("\n\n BOTON PRESIONADO DE LA
ALIMENTACION LIQUIDA \n ")
archivo_salida.write(" FECHA Y HORA DEL CONSUMO __ DD/MM/AA
<==> HH/MM/SS: ")
archivo_salida.write (imprimir _ fecha ())
archivo_salida.close()
def LED3():
    arduino.write('T')
    label41 = Label(root, text="**ENCENDIDO**", font=("gothica", 11),bg="sky
blue", fg="dark green")
    label41.grid(row=8, column=4)
    button15 = Button(root, text="  ",bg="green",font=("Helvetica", 10))
    button15.grid(row=8, column=2)
    button3.configure(text=" PROCESANDO ",state=DISABLED)
def LED4():
    arduino.write('C')
    label42= Label(root, text="**ENCENDIDO**", font=("gothica", 11),bg="sky
blue", fg="dark green")
    label42.grid(row=10, column=4)
    button16 = Button(root, text="  ",bg="green",font=("Helvetica", 10))
    button16.grid(row=10, column=2)
    button4.configure(text=" PROCESANDO ",state=DISABLED)
def LUZP():

```



```

    arduino.write('P')
    label31= Label(root, text="LUZ*APAGADA ", font=("gothica", 11), bg="sky
blue",fg="black")
    label31.grid(row=8, column=4)
    button9 = Button(root, text="  ",bg="red",font=("Helvetica", 10))
    button9.grid(row=8, column=2)
    button3.configure(text=" LUZ DE LOS ALIMENTOS",state=NORMAL)
def LUZE():
    arduino.write('Q')
    label32= Label(root, text="LUZ*APAGADA ", font=("gothica", 11),bg="sky
blue", fg="black")
    label32.grid(row=10, column=4)
    button10 = Button(root, text="  ",bg="red",font=("Helvetica", 10))
    button10.grid(row=10, column=2)
    button4.configure(text=" LUZ DE EXTERIORES",state=NORMAL)
def play():
    arduino.write('M')
    time.sleep(1)
    arduino.write('R')
def next():
    arduino.write('N')
    time.sleep(1)
    arduino.write('E')
def reset():
    arduino.write('S')
    time.sleep(1)
    arduino.write('A')
def retroceder():
    arduino.write('K')
    time.sleep(1)
    arduino.write('H')

```



```

    archivo_salida.write(" VALOR ACTUAL DEL CONSUMO EN (LITROS):
")
    archivo_salida.write(sensor)
    archivo_salida.write(" FECHA Y HORA DE CONSUMO __ DD/MM/AA
<==> HH/MM/SS: ")
    archivo_salida.write (imprimir _ fecha ())
    archivo_salida.close()
    label34= Label(root, text="PRESIONE*BOTON*L", font=("gothica", 12),
bg="sky blue",fg="black")
    label34.grid(row=6, column=6)
    button8 = Button(root, text=" ",bg="red",font=("Helvetica", 10))
    button8.grid(row=6, column=4)
def solido1():
    arduino.open()
    arduino.write('O')
    sensor = arduino.readline()
    label45 = Label(root, text=sensor,bg="sky blue",font=("gothica", 12))
    label45.grid(row=22, column=6)
def liquido1():
    arduino.open()
    arduino.write('I')
    sensor = arduino.readline()
    label46 = Label(root, text=sensor, bg="sky blue",font=("gothica", 12))
    label46.grid(row=22, column=8)
def imprimir _ fecha ():
    return str(time.localtime()[2]) + "/" + str(time.localtime()[1]) + "/" +
str(time.localtime()[0]) + ", " + str(time.localtime()[3]) + ":" +
str(time.localtime()[4]) + ":" + str(time.localtime()[5])
    root = Tk()
    button1 = Button(root, text=" ALIMENTACION SOLIDA
",fg="purple",font=("Helvetica", 10),relief=SOLID, command=LED1)

```

```

button1.grid(row=4, column=8)
button2 = Button(root, text=" ALIMENTACION LIQUIDA", fg="navy
blue",font=("Helvetica", 10),relief=SOLID,command=LED2)
button2.grid(row=6, column=8)
button3 = Button(root, text=" LUZ DE LOS
ALIMENTOS",fg="blue",font=("Helvetica", 10),relief=SOLID,
command=LED3)
button3.grid(row=8, column=8)
button4 = Button(root, text=" LUZ DE EXTERIORES
",fg="brown",font=("Helvetica", 10),relief=SOLID, command=LED4)
button4.grid(row=10, column=8)
button5= Button(root, text="APAGAR LUZ PERRO
",fg="red",font=("Helvetica", 10),relief=SOLID, command=LUZP)
button5.grid(row=8, column=6)
button6= Button(root, text="APAGAR LUZ EXTERIOR
",fg="red",font=("Helvetica", 10),relief=SOLID, command=LUZE)
button6.grid(row=10, column=6)
button7= Button(root, text=" ",bg="red",font=("Helvetica", 10))
button7.grid(row=4, column=4)
button8= Button(root, text=" ",bg="red",font=("Helvetica", 10))
button8.grid(row=6, column=4)
button9 = Button(root, text=" ",bg="red",font=("Helvetica", 10))
button9.grid(row=8, column=2)
button10= Button(root, text=" ",bg="red",font=("Helvetica", 10))
button10.grid(row=10, column=2)
button11= Button(root, text=" PLAY/PAUSA ", fg="red",font=("Helvetica",
10),relief=SOLID,command=play)
button11.grid(row=14, column=8)
button12= Button(root, text="INICIAR/RESET", fg="blue",font=("Helvetica",
10),relief=SOLID,command=reset)
button12.grid(row=16, column=8)

```

```

button13= Button(root, text=" SIGUIENTE >> ",fg="dark
green",font=("Helvetica", 10),relief=SOLID, command=next)
button13.grid(row=14, column=6)
button14= Button(root, text="RETROCEDER <<",
fg="purple",font=("Helvetica", 10),relief=SOLID,command=retroceder)
button14.grid(row=16, column=6)
hi_there1= Button(root, text="CONSUMO *S*",fg="dark
green",font=("Helvetica", 10),relief=SOLID, command=solido)
hi_there1.grid(row=20, column=2)
hi_there2= Button(root, text="CONSUMO *L*",fg="dark
green",font=("Helvetica", 10),relief=SOLID, command=liquido)
hi_there2.grid(row=20, column=4)
hi_there3= Button(root, text="RESTANTE *S*",fg="dark
green",font=("Helvetica", 10),relief=SOLID, command=solido1)
hi_there3.grid(row=20, column=6)
hi_there4= Button(root, text="RESTANTE *L*",fg="dark
green",font=("Helvetica", 10),relief=SOLID, command=liquido1)
hi_there4.grid(row=20, column=8)
label1 = Label(root, text=" ",bg="sky blue", font=("Helvetica", 10))
label1.grid(row=1, column=1)
label2 = Label(root, text=" ",bg="sky blue", font=("Helvetica", 10))
label2.grid(row=3)
label3= Label(root, text=" ",bg="sky blue", font=("Helvetica", 10))
label3.grid(row=5)
label4= Label(root, text=" ",bg="sky blue",font=("Helvetica", 10))
label4.grid(row=7)
label5= Label(root, text=" ",bg="sky blue", font=("Helvetica", 10))
label5.grid(row=9)
label6= Label(root, text=" ",bg="sky blue",font=("Helvetica", 5))
label6.grid(row=11)
label7 = Label(root, text=" ",bg="sky blue",font=("Helvetica", 5))

```

```

label7.grid(row=13)
label8 = Label(root, text="    ",bg="sky blue",font=("Helvetica", 5))
label8.grid(row=15)
label9 = Label(root, text="    ",bg="sky blue",font=("Helvetica", 5))
label9.grid(row=17)
label10 = Label(root, text="    ",bg="sky blue",font=("Helvetica", 5))
label10.grid(row=19)
label11 = Label(root, text="    ",bg="sky blue",font=("Helvetica", 5))
label11.grid(row=21)
label12 = Label(root, text="    ",bg="sky blue",font=("Helvetica", 10))
label12.grid(column=3)
label13 = Label(root, text="    ",bg="sky blue",font=("Helvetica", 10))
label13.grid(column=5)
label14 = Label(root, text="    ",bg="sky blue",font=("Helvetica", 10))
label14.grid(column=7)
label15 = Label(root, text="    ",bg="sky blue",font=("Helvetica", 10))
label15.grid(column=9)
label16 = Label(root, text="CONTROL",bg="sky blue", fg="blue",
font=("Helvetica", 20))
label16.grid(row=2, column=4)
label17 = Label(root, text=" Y ",bg="sky blue", fg="blue",font=("Helvetica", 20))
label17.grid(row=2, column=5)
label18 = Label(root, text="MONITOREO",bg="sky blue",fg="blue",
font=("Helvetica", 20))
label18.grid(row=2, column=6)
label19 = Label(root, text="DE",bg="sky blue",fg="blue", font=("Helvetica", 20))
label19.grid(row=2, column=7)
label20 = Label(root, text="CANES",bg="sky blue",fg="blue", font=("Helvetica",
20))
label20.grid(row=2, column=8)
label21 = Label(root, text="CONTROL",bg="sky blue",font=("Helvetica", 15))

```

```

label21.grid(row=18,column=2)
label22 = Label(root, text=" DEL",bg="sky blue",font=("Helvetica", 15))
label22.grid(row=18,column=3)
label23= Label(root, text="CONSUMO",bg="sky blue",font=("Helvetica", 15))
label23.grid(row=18,column=4)
label24 = Label(root, text=" DE LA",bg="sky blue",font=("Helvetica", 15))
label24.grid(row=18,column=5)
label25 = Label(root, text="ALIMENTACION",bg="sky blue",font=("Helvetica",
15))
label25.grid(row=18,column=6)
label26 = Label(root, text="*S* Y *L*",bg="sky blue",font=("Helvetica", 15))
label26.grid(row=18,column=7)
label27 = Label(root, text="DEL CAN",bg="sky blue",font=("Helvetica", 15))
label27.grid(row=18,column=8)
label28 = Label(root, text="CONTROL",bg="sky blue",fg="brown",
font=("Helvetica", 15))
label28.grid(row=12, column=6)
label29 = Label(root, text="DE",bg="sky blue",fg="brown",font=("Helvetica",
15))
label29.grid(row=12, column=7)
label30 = Label(root, text="AUDIO",bg="sky
blue",fg="brown",font=("Helvetica", 15))
label30.grid(row=12, column=8)
label31 = Label(root, text="LUZ*APAGADA", font=("gothica", 11), bg="sky
blue",fg="black")
label31.grid(row=8, column=4)
label32 = Label(root, text="LUZ*APAGADA", font=("gothica", 11), bg="sky
blue",fg="black")
label32.grid(row=10, column=4)
label33= Label(root, text="PRESIONE*BOTON*S", font=("gothica", 12),
bg="sky blue",fg="black")

```

```
label33.grid(row=4, column=6)
label34= Label(root, text="PRESIONE*BOTON*L", font=("gothica", 12),
bg="sky blue",fg="black")
label34.grid(row=6, column=6)
label35 = Label(root, text=cero,bg="sky blue",font=("Helvetica", 12))
label35.grid(row=22, column=2)
label36 = Label(root, text=cero,bg="sky blue",font=("Helvetica", 12))
label36.grid(row=22, column=4)
label37= Label(root, text=cero,bg="sky blue",font=("Helvetica", 12))
label37.grid(row=22, column=6)
label38 = Label(root, text=cero,bg="sky blue",font=("Helvetica", 12))
label38.grid(row=22, column=8)
root.title('ALIMENTACION DEL CAN')
root.geometry("800x600")
root.config(bg="sky blue")
root.mainloop()
```


ANEXO E

Programación de la Tarjeta Electrónica Arduino

Programación de la Tarjeta Electrónica Arduino

A continuación se presenta la programación en Arduino Uno:

```
int i = 0;
float aux;
float aux1;
float val = 0;
float val1 = 0;
float valor = 0;
float valor1 = 0;
#define leduno 3
#define leddos 4
#define ledtres 5
#define ledcuatro 6
#define play 7
#define next 8
#define reset 9
#define retroceder 10
void setup() { {
  Serial.begin(9600);
  for (i=3; i<=10; i++){
    pinMode(i, OUTPUT);
  }
  pinMode(12, OUTPUT);
  pinMode(11, OUTPUT);
}
void loop() {
  {
  if(Serial.available()) {
    char in = Serial.read();
    if(in == 'U'){
```

```

    digitalWrite(leduno, HIGH);
}
if(in == 'D'){
    digitalWrite(leddos, HIGH);
}
if(in == 'T'){
    digitalWrite(ledtres, HIGH);
}
if(in == 'C'){
    digitalWrite(ledcuatro, HIGH);
}
if(in == 'P'){
    digitalWrite(ledtres, LOW);
}
if(in == 'Q'){
    digitalWrite(ledcuatro, LOW);
}
if(in == 'X'){
    digitalWrite(leduno, LOW);
}
if(in == 'Y'){
    digitalWrite(leddos, LOW);
}
if(in == 'M'){
    digitalWrite(play, LOW);
}
if(in == 'R'){
    digitalWrite(play, HIGH);
}
if(in == 'N'){
    digitalWrite(next, LOW);
}

```

```

}
if(in == 'E'){
    digitalWrite(next, HIGH);
}
if(in == 'S'){
    digitalWrite(reset, LOW);
}
if(in == 'A'){
    digitalWrite(reset, HIGH);
}
if(in == 'K'){
    digitalWrite(retroceder, LOW);
}
if(in == 'H'){
    digitalWrite(retroceder, HIGH);
}
if(in == 'Z'){
    valor = (valor + (val-7.33));
    Serial.println(valor);
}
if(in == 'O'){
    aux = (10 - valor);
    Serial.println(aux);
}
if(in == 'B'){
    valor1 = (valor1 + (val1-26.88));
    Serial.println(valor1);
}
if(in == 'I'){
    aux1 = (100 - valor1);
    Serial.println(aux1);
}

```

```
    }  
  }  
}  
  //alimentación de sólidos  
  val = (5.0 * analogRead(0)*100.0)/1023.0;  
  if (val > 7){  
    digitalWrite(12,LOW);  
  }else  
    digitalWrite(12,HIGH);  
  }  
  //alimentación de líquido  
  val1 = (5.0 * analogRead(1)*100.0)/1023.0;  
  if (val1 > 44){  
    digitalWrite(11,LOW);  
  }else  
    digitalWrite(11,HIGH);  
  }  
}
```

ANEXO F
CÓDIGO DE PROGRAMACIÓN DE LAS DOS
CÁMARAS

CÓDIGO DE PROGRAMACIÓN DE LAS DOS CÁMARAS

Programación.

Realizamos el siguiente comando `sudo nano /var/www/camarasweb.html` para teclear dentro de este el siguiente código realizado en php y HTML.

```
<!-- THREE STEPS TO INSTALL WEB CAM TIME:
```

1. Copy the coding into the HEAD of your HTML document
2. Add the onLoad event handler into the BODY tag
3. Put the last coding into the BODY of your HTML document -->

```
<!-- STEP ONE: Paste this code into the HEAD of your HTML document -->
```

```
<HEAD>
```

```
<SCRIPT LANGUAGE="JavaScript">
```

```
<!-- This script and many more are available free online at -->
```

```
<!-- The JavaScript Source!! http://javascript.internet.com -->
```

```
</script>
```

```
</HEAD>
```

```
<!-- STEP TWO: Insert the onLoad event handler into your BODY tag -->
```

```
<BODY BGCOLOR=#93DB70>
```

```
<!-- STEP THREE: Copy this code into the BODY of your HTML document -->
```

```
<table border=10 width="320" align=right BGCOLOR=#C0C0C0>
```

```
<tr align=center valign=center>
```

```
<td align=center valign=center width="320" height="240">
```

```
<SCRIPT LANGUAGE="JavaScript">
```

```
<!-- Original: Chris Herrod (christopher\_herrod@hotmail.com) -->
```

```
<!-- Web Site: http://members.nbci.com/chris\_herrod -->
```

```
<!-- This script and many more are available free online at -->
```

```
<!-- The JavaScript Source!! http://javascript.internet.com -->
```

```
<!-- Begin
```

```
var address ="" ;
```

```
address = "http://192.168.1.109:8082";
```

```
document.write("<img src="" + address + "">");
```

```
// End -->
```

```

</script>
</td>
</tr>
<tr align=center valign=center>
<td align=center valign=center width="320">
<form>
<input size=30 value="VIGILANCIA EXTERNA DEL CAN">
</form>
</table>
<HEAD>
<SCRIPT LANGUAGE="JavaScript">
<!-- This script and many more are available free online at -->
<!-- The JavaScript Source!! http://javascript.internet.com -->
</script>
</HEAD>
<!-- STEP TWO: Insert the onLoad event handler into your BODY tag -->
<BODY onLoad="Reload();startclock()">
<!-- STEP THREE: Copy this code into the BODY of your HTML document -->
<table border=10 width="320" BGCOLOR=#A68064>
<tr align=center valign=center>
<td align=center valign=center width="320" height="240">
<SCRIPT LANGUAGE="JavaScript">
<!-- Original: Chris Herrod (christopher_herrod@hotmail.com) -->
<!-- Web Site: http://members.nbc.com/chris\_herrod -->
<!-- This script and many more are available free online at -->
<!-- The JavaScript Source!! http://javascript.internet.com -->
<!-- Begin
var address ="" ;
address = "http://192.168.1.109:8081";
document.write("<img src="" + address + "">");
// End -->

```



```
</script>
</td>
</tr>
<tr align=center valign=center>
<td align=center valign=center width="320">
<form>
<input value="VIGILANCIA DE ALIMENTACION">
</form>
</table>
<p><center>
<font face="gothica" size=10 COLOR="red">VIDEO VIGILANCIA<br>
</center><p>
<!-- Script Size: 2.95 KB -->
```

ANEXO G
MANUAL DE USUARIO PARA LA ALIMENTACIÓN
DEL CAN

MANUAL DE USUARIO PARA LA ALIMENTACIÓN DEL CAN

1. Conexión y Encendido de los Equipos a Utilizar.

Previamente el usuario debe energizar los aparatos a funcionar y encender la computadora que estará en conexión de área local con la mini computadora Raspberry Pi estos equipos puede estar conectado mediante el cable de (red RJ45) o mediante comunicación inalámbrica (Wifi), todo esto en Área Local.

Router con modem conectado a internet.



Figura G.1 Conexión en área local con Raspberry y Computadora personal.

2. Conexión desde el internet con TeamViewer.

La computadora que estará conectada al Raspberry Pi de forma remota, debe estar inicializado el programa TeamViewer para que tenga una conexión exitosa.

Una vez instalado el programa TeamViewer, desde el lugar que vayamos a establecer una comunicación con la otra computadora que estará conectado al Raspberry Pi de forma remota procedemos a correr el programa TeamViewer.

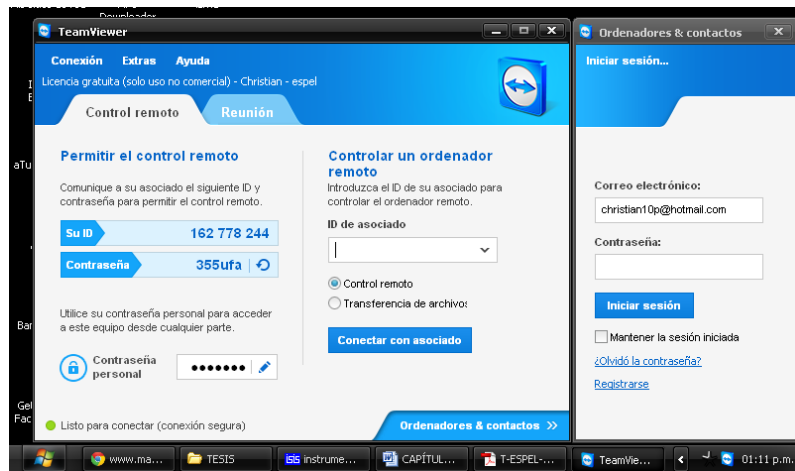


Figura G.2 Ventana del programa TeamViewer.

Primero introducimos su ID de asociado 867102465 y la contraseña personal establecida *espe12013* y procedemos a conectar con la otra computadora de forma remota.

Una vez establecido la conexión con la otra computadora y obtener el escritorio del mismo procedemos a buscar el programa Xming que se encuentra en el escritorio.



Figura G.3 Corrido del programa TeamViewer en Escritorio Remoto.

3. Ingrese en acceso remoto al raspberry pi con el programa Xming

Procedemos a correr el programa Xming para obtener el siguiente cuadro, donde se introdujera la contraseña *espe12013* y proceder hacer un escritorio remoto con el raspberry pi mediante el programa Xming.

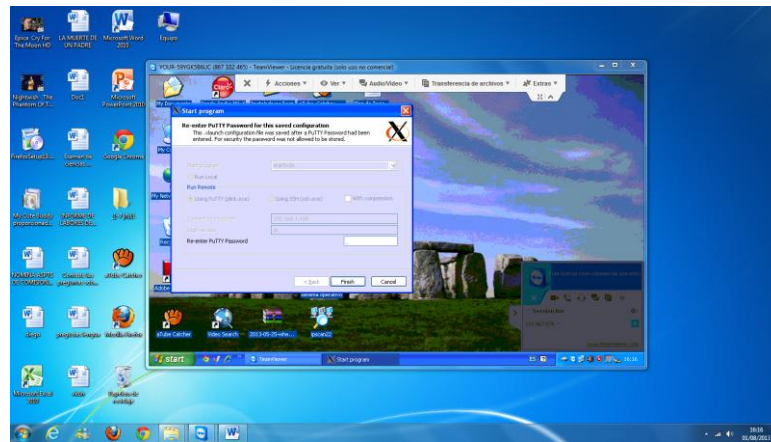


Figura G.4 Corrido del programa Xming para acceder al Escritorio Remoto del Raspberry Pi.

Una vez introducido la contraseña mediante el programa Xming obtenemos el escritorio del Raspberry Pi:

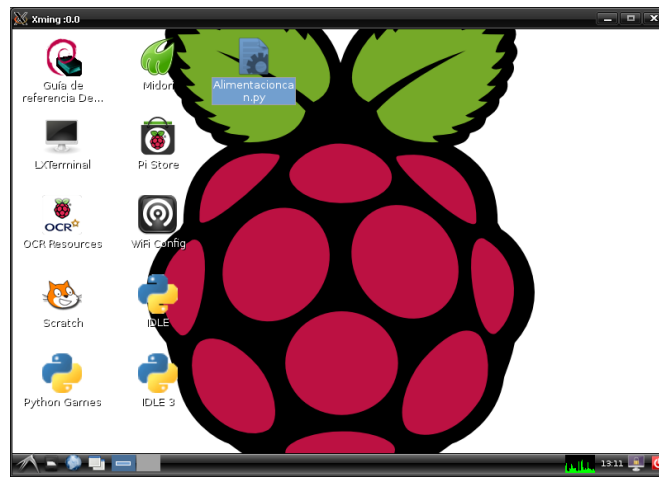


Figura G.5 Escritorio Remoto del Raspberry Pi.

4. Inicio del explorador de internet en Raspberry Pi para utilizar las cámaras.

Al obtener el escritorio remoto del raspberry pi, nos dirigimos a un explorador de internet en nuestro caso *chromium* que se encuentra en la barra de inicio del escritorio del raspberry pi.

Inicializado el explorador chromium procedemos a introducir la dirección de las cámaras web que es : 192.168.1.109/camarasweb.html y obtendremos las dos cámaras web en tiempo real.

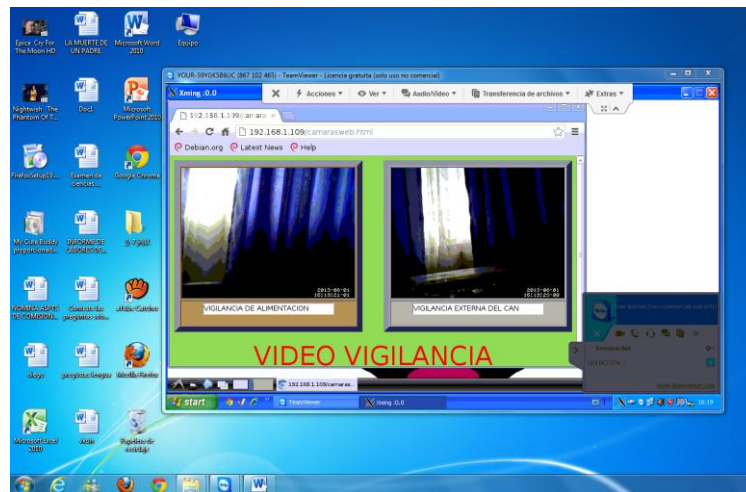


Figura G.7 Visualización de las cámaras por Internet por medio de Acceso Remoto.

5. Ingreso al programa de alimentación de los animales.

Ingresamos al escritorio del Raspberry Pi, posteriormente abrimos el programa de alimentacióncan.py en el escritorio donde correremos el programa haciendo doble clic en el mismo:

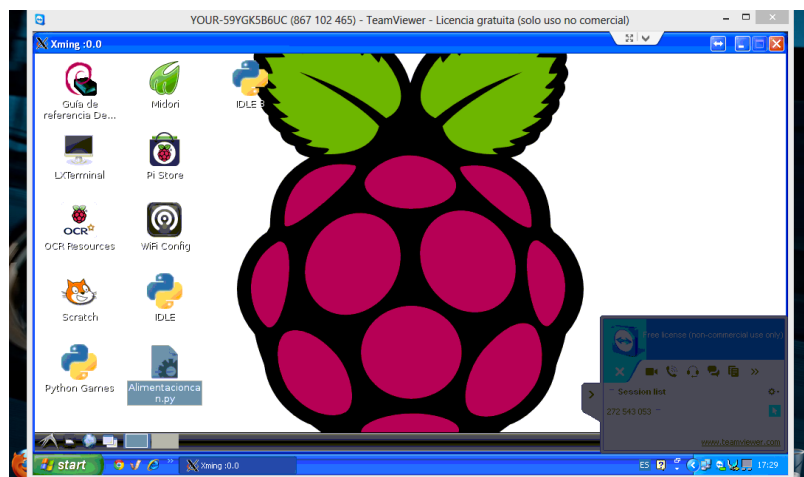


Figura G.8 Escritorio del Raspberry donde correremos el Programa Alimentacióncan.py.

Al realizar doble clic en nuestro programa alimentacioncan.py nos indicara una ventana donde escogeremos la opción EJECUTAR y se abrirá la ventana con nuestro panel principal.



Figura G.9 Ventana donde se escoge la opción Ejecutar.

Cuando el programa de Python este corriendo se aparecerá una nueva ventana con los botones que controlan el proceso de alimentación.



Figura G.10 Ventana del proceso de alimentación.

6. Visualización de las dos ventanas, cámaras como el panel de alimentación.

En la siguiente figura observamos una ventana con las cámaras web y la ventana con los botones del proceso de alimentación, se debe destacar que cada ventana es individual tanto como el de las cámaras como la ventana del proceso de alimentación.

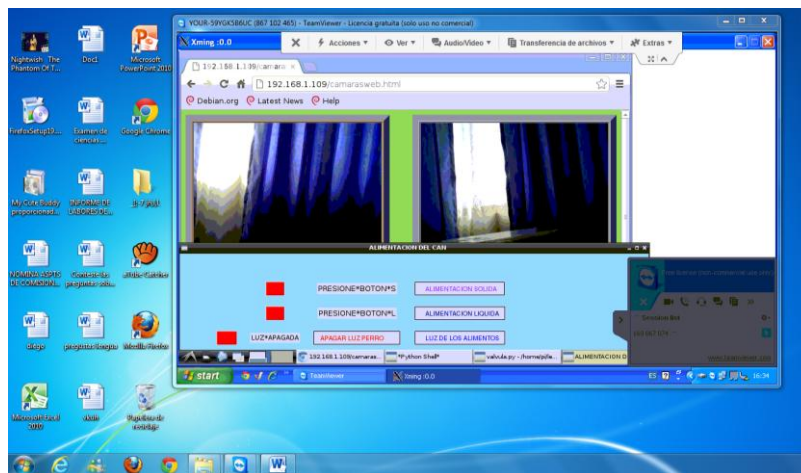


Figura G.11 Ventanas del proceso de Alimentación y Vigilancia, observado desde el Internet por Acceso Remoto con TeamViewer.

7. Utilización del panel de alimentación para realizar el proceso de alimentación.

A continuación se detalla cada uno de los botones presentes en este panel de control.

• Alimentación Sólida:

En esta parte encontramos el botón de “Alimentación Sólida” y un aviso que nos indica que debemos presionar el botón de alimentación sólida “PRESIONE*BOTON*S”, para que este entre en funcionamiento y además tenemos un indicador luminoso que cuando está en rojo nos indica que el proceso está apagado.

Si se escoge la opción “Alimentación sólida” y se pulsa el botón mencionado, estamos enviando una señal para que se active el motor de la banda transportadora de comida y este dirija el alimento al recipiente adecuado.

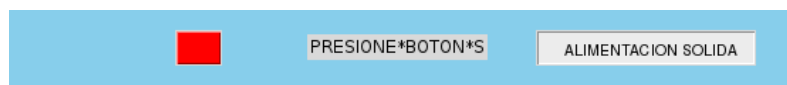


Figura G.12 Botones en estado de Apagado de la Alimentación Sólida.

Cuando se ha realizado la activación del botón de alimentación sólida, el Sensor entrara en funcionamiento hasta que este detecte el peso deseado de la alimentación y terminado el proceso, el aviso se pone en ****FINALIZADO**** y el indicador de luminosidad se activa de color verde con este proceso se ha indicado que la alimentación sólida ha terminado.



Figura G.13 Botones en estado de Encendido de la Alimentación Sólida

• **Alimentación Líquida:**

En esta parte encontramos el botón de “Alimentación Líquida” y un aviso que nos indica que debemos presionar el botón de alimentación líquida “PRESIONE*BOTON*L”, para que este entre en funcionamiento y además tenemos un indicador luminoso que cuando está en rojo nos indica que el proceso está apagado.

Si se escoge la opción “Alimentación Líquida” y se pulsa en el botón, estamos enviando una señal para que se active la electroválvula y este dirija el alimento líquido al recipiente adecuado.

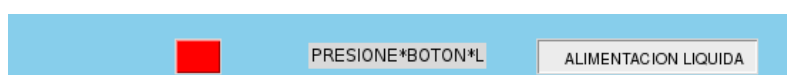


Figura G.14 Botones en estado de Apagado de la Alimentación Líquida.

Cuando se ha realizado la activación del botón de alimentación líquida, el Sensor entrara en funcionamiento hasta que este detecte el peso deseado de la alimentación líquida y terminado el proceso, el aviso se pone en ****FINALIZADO**** de color verde como el indicador luminoso que también se pone de color verde.



Figura G.15 Botones en estado de Encendido de la Alimentación Líquida.

• **Luz de los Alimentos:**

Con el siguiente botón podemos brindar luz en la parte de alimentación en caso de que el usuario tenga que observar y brindar la alimentación a sus animales en la noche.

La activación está compuesta por un botón que dice “LUZ DE LOS ALIMENTOS” y el apagado del mismo está por otro botón que dice “APAGAR LUZ PERRO”, un aviso que nos indica “LUZ APAGADA” y un indicador luminoso que esta de color rojo cuando está apagado



Figura G.16 Botones en estado de Apagado de la Luz de los Alimentos.

Cuando el proceso esta encendido se indicara en palabras “ENCENDIDO” y el indicador luminoso se mostrara de color verde, posteriormente se apagara con el botón “APAGAR LUZ PERRO”.

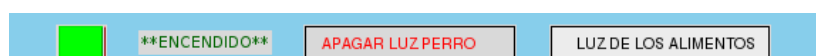


Figura G.17 Botones en estado de Encendido de la Luz de los Alimentos.

- **Luz de Exteriores:**

Con el siguiente botón podemos brindar luz en la parte de los exteriores en caso de que el usuario tenga que observar y vigilar mediante la cámara a sus animales en la noche.

La activación está compuesta por un botón que dice “LUZ DE EXTERIORES” y el apagado del mismo está por otro botón que dice “APAGAR LUZ EXTERIOR”, un aviso que nos indica “LUZ APAGADA” y un indicador luminoso que esta de color rojo cuando está apagado



Figura G.18 Botones en estado de Apagado de la Luz de Exteriores.

Cuando el proceso esta encendido se indicara en palabras “ENCENDIDO” y el indicador luminoso se mostrara de color verde, posteriormente se apagara con el botón “APAGAR LUZ EXTERIORES”.



Figura G.19 Botones en estado de Encendido de la Luz de Exteriores.

- **Control de audio para el llamado del animal:**

Con los siguientes botones se puede hacer reproducir los diferentes tonos o llamados al animal cuando estos estén pre grabados en el dispositivo de audio wtv020sd16p.



Figura G.20 Botones de audio.

- **Control de registro de consumo y verificación en la alimentación del can.**

Con los siguientes botones realizamos una verificación del consumo del alimento suministrado al can y podemos saber cuánto de comida nos resta del total de la comida y los datos se quedan guardados en un registro para posteriores consulta.

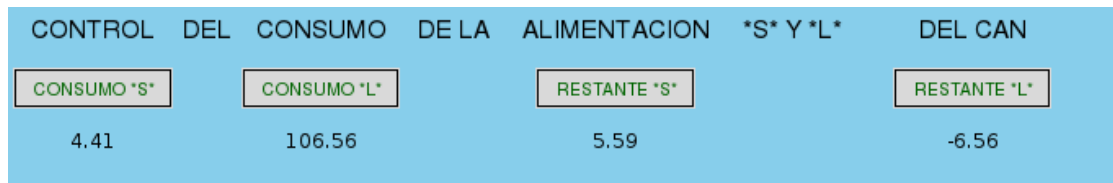


Figura G.21 Botones de registro de alimentos.

8. Verificación de registro de consumo de la Alimentación del can.

En la siguiente figura G.22 se indica la barra de inicio donde ingresaremos al gestor de archivos del Raspberry Pi donde encontraremos los registros de consumo de los alimentos.



Figura G.22 Barra de Gestor de archivos del Raspberry Pi.

Una vez ingresado al gestor de archivos del Raspberry Pi, encontramos los archivos de consumo y registro de alimentación en formato txt.

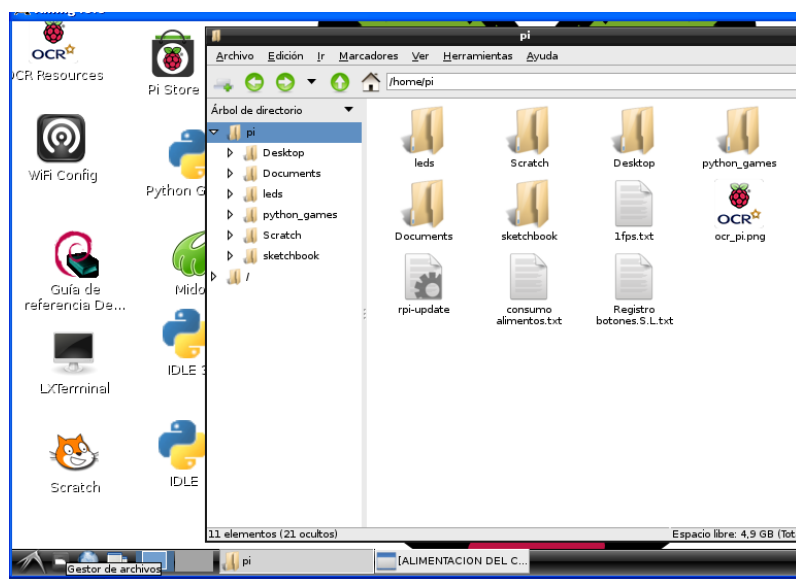
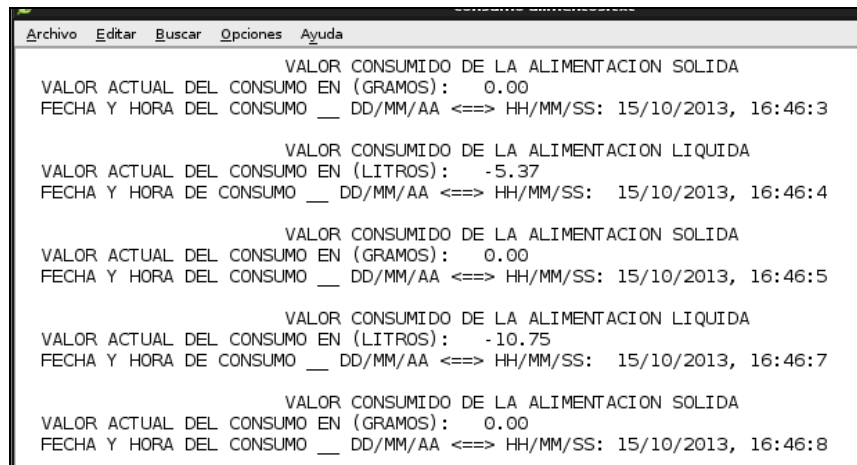


Figura G.23 Ventana de gestor de archivos del Raspberry Pi.

9. Prueba en el sistema de registro de consume de los alimentos.

En la figura G.24 se presenta un ejemplo realizado en el sistema de registro de consumo de la alimentación Sólida y Liquida, la cual se verifica que se guardan los datos ingresados del sensor; es decir es una pantalla de consultas.



```
Archivo  Editar  Buscar  Opciones  Ayuda
VALOR CONSUMIDO DE LA ALIMENTACION SOLIDA
VALOR ACTUAL DEL CONSUMO EN (GRAMOS):  0.00
FECHA Y HORA DEL CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:46:3

VALOR CONSUMIDO DE LA ALIMENTACION LIQUIDA
VALOR ACTUAL DEL CONSUMO EN (LITROS):  -5.37
FECHA Y HORA DE CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:46:4

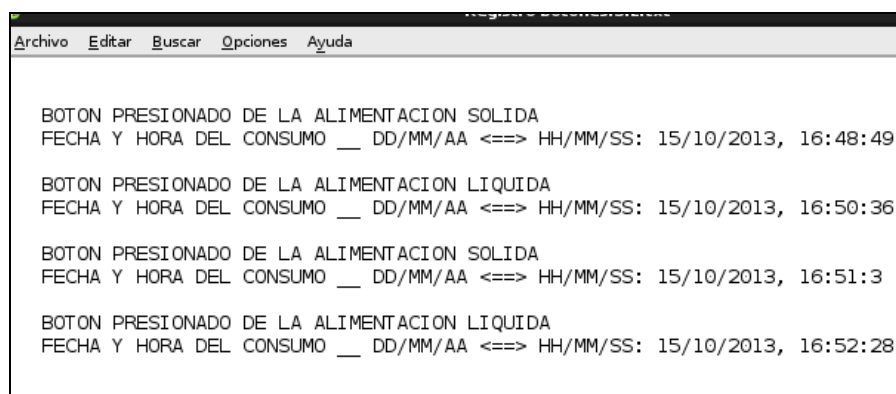
VALOR CONSUMIDO DE LA ALIMENTACION SOLIDA
VALOR ACTUAL DEL CONSUMO EN (GRAMOS):  0.00
FECHA Y HORA DEL CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:46:5

VALOR CONSUMIDO DE LA ALIMENTACION LIQUIDA
VALOR ACTUAL DEL CONSUMO EN (LITROS):  -10.75
FECHA Y HORA DE CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:46:7

VALOR CONSUMIDO DE LA ALIMENTACION SOLIDA
VALOR ACTUAL DEL CONSUMO EN (GRAMOS):  0.00
FECHA Y HORA DEL CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:46:8
```

Figura G.24 Verificación de datos guardados de la Alimentación Sólida y Liquida.

En la figura G.25 se presenta un ejemplo realizado en el sistema de registro de la alimentación Sólida y Liquida, la cual se verifica que se guardan los datos ingresados cuando se presiona el botón que realiza la alimentación sólida y líquida; es decir es una pantalla de consultas.



```
Archivo  Editar  Buscar  Opciones  Ayuda
BOTON PRESIONADO DE LA ALIMENTACION SOLIDA
FECHA Y HORA DEL CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:48:49

BOTON PRESIONADO DE LA ALIMENTACION LIQUIDA
FECHA Y HORA DEL CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:50:36

BOTON PRESIONADO DE LA ALIMENTACION SOLIDA
FECHA Y HORA DEL CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:51:3

BOTON PRESIONADO DE LA ALIMENTACION LIQUIDA
FECHA Y HORA DEL CONSUMO __ DD/MM/AA <==> HH/MM/SS: 15/10/2013, 16:52:28
```

Figura G.25 Verificación de datos guardados al presionar el botón de la Alimentación Sólida y Liquida.

10. Salida y cierre de la Alimentación del Can.

Al terminar el proceso de Alimentación del can procedemos a cerrar la ventana de las cámaras, los registros de consumo y la ventana que está el panel del proceso de Alimentación la dejamos abierta para posteriores control de la alimentación, así obtenemos un registro permanente del sistema de alimentación y no perdemos datos importantes en el panel frontal del proceso.

A continuación salimos del programa TeamViewer dejando abierta la ventana de del panel del proceso de alimentación y posterior cerrar el programa de TeamViewer así terminar toda la comunicación con el proceso de alimentación.

ANEXO H
Manual de operador.

Manual de operador.

Descripción.

El proceso está diseñado para un solo valor de comida sólida y líquida de una cierta edad y peso del animal.

Para posible crecimiento y peso del animal se debe realizar cambios en la programación del panel frontal de Raspberry Pi como en la programación de Arduino Uno, por los sensores que deben medir diferentes valores de peso según el usuario crea necesario.

Programación en Raspberry PI

En la siguiente tabla H.1 se indicara los valores en la alimentación sólida, líquida y valores que se deben colocar en la programación de Python en Raspberry Pi y en la placa Arduino uno.

Tabla H.1 Valores para una alimentación Sólida

| Valores para una Alimentación Sólida del Animal. | | | |
|---------------------------------------------------------|--------------------|-----------------|----------------|
| Valor (Python) | Cantidad en libras | Cantidad Gramos | Cantidad Tazas |
| 5 | 0.25 | 113.40 | 1/2 |
| 10 | 0.5 | 226.8 | 1 |
| 15 | 0.75 | 340.19 | 1 1/2 |
| 20 | 1 | 453.59 | 2 |

Tabla H.2 Valores para una alimentación Líquida

| Valores para una Alimentación Líquida del Animal | | |
|---------------------------------------------------------|-----------------|---------------------|
| Valor (Python) | Valor en Litros | Valor en mililitros |
| 15 | 0.25 | 250 |
| 30 | 0.5 | 500 |
| 45 | 0.75 | 750 |
| 60 | 1 | 1000 |

Programación en Python con los nuevos valores de la Alimentación del Animal.

Ingresamos al escritorio del Raspberry Pi y abrimos el programa desarrollado en Python para cambiar los valores deseados.

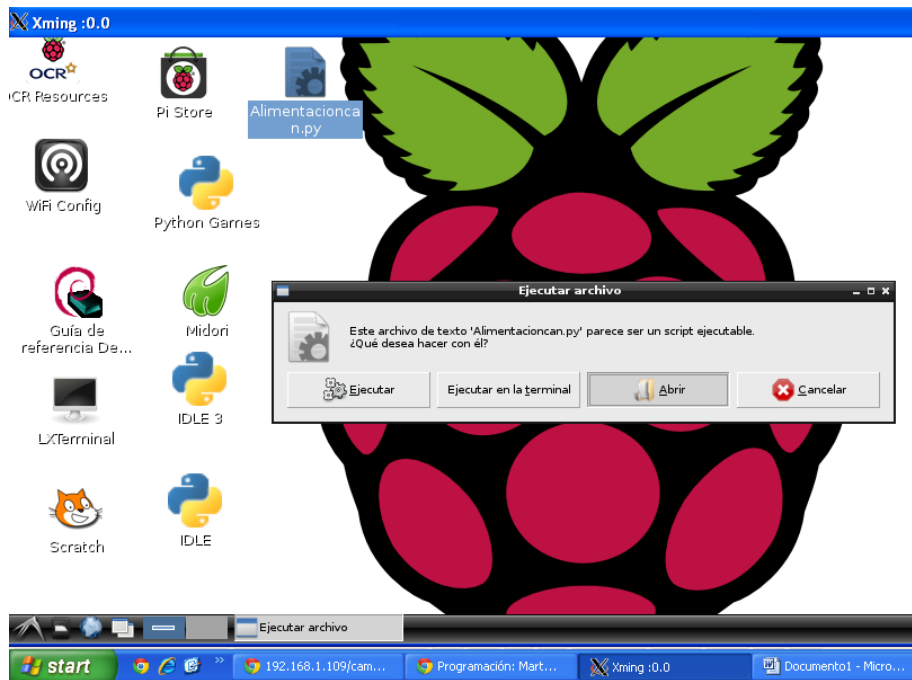


Figura H.1: Ventana donde se escoge la opción Abrir.

Abierto el programa realizado en Python, nos dirigimos a la línea que dice defLED1():

Y al comando time.sleep(10) para cambiar y colocar el valor deseado por el usuario según la tabla de valores de alimentación Sólida.

```
defLED1(){
    arduino.write('U')
    time.sleep(10)
    button17= Button(root, text=" ",bg="green",font=("Helvetica", 10))
    button17.grid(row=4, column=4)
    label39 = Label(root, text="*****FINALIZADO*****", font=("gothica", 12),
    bg="sky blue",fg="dark green")
    label39.grid(row=4, column=6)
    arduino.write('X')
    archivo_salida = open("Registro boton S.txt","a")
    archivo_salida.write("\n\n BOTON PRESIONADO DE LA
ALIMENTACION SOLIDA \n ")
    archivo_salida.write(" FECHA Y HORA DEL CONSUMO __ DDMM/AA
<==> HHMM/SS: ")
```

Figura H.2 Programación en Python de la Alimentación Sólida.

Después de realizar la alimentación sólida nos dirigimos a otra línea para cambiar el valor de la alimentación líquida.

Colocados en el programa nos dirigimos a la línea que dice defLED2(): y al comando time.sleep(82) para cambiar y colocar el valor deseado por el usuario según la tabla de valores de la alimentación líquida.

```
def LED2():
    arduino.write('D')
    time.sleep(82)
    button18= Button(root, text=" ",bg="green",font=("Helvetica", 10))
    button18.grid(row=6, column=4)
    label40 = Label(root, text="*****FINALIZADO*****", font=("gothica",
12),bg="sky blue", fg="dark green")
    label40.grid(row=6, column=6)
    arduino.write('Y')
    archivo_salida = open("Registro boton L.txt","a")
    archivo_salida.write("\n\n BOTON PRESIONADO DE LA
ALIMENTACION LIQUIDA \n ")
    archivo_salida.write(" FECHA Y HORA DEL CONSUMO __ DD/MM/AA
<==> HH/MM/SS: ")
    archivo_salida.write(imprimir_fecha())
```

Figura H.3 Programación en Python de la Alimentación Líquida.

Al terminar de realizar estos cambios procedemos a ir a la barra de herramientas a FILE y a guardar el programa realizado en Python.

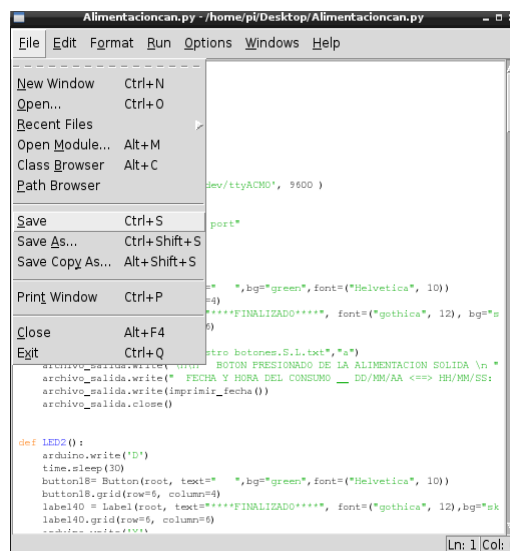


Figura H.4 Ventana de Python en herramientas y Guardar.

Realizado el guardado del programa en Python cerramos la ventana.

Programación en Arduino Uno

En la siguiente tabla H.3 se indicara los valores en la alimentación sólida, líquida y valores que se deben colocar en la programación en la placa Arduino uno.

Tabla H.3 Valores para una alimentación Sólida

| Valores para una Alimentación Sólida del Animal. | | | |
|---------------------------------------------------------|--------------------|-----------------|----------------|
| Valor (Python) | Cantidad en libras | Cantidad Gramos | Cantidad Tazas |
| 3.5 | 0.25 | 113.40 | 1/2 |
| 7 | 0.5 | 226.8 | 1 |
| 10.5 | 0.75 | 340.19 | 1 1/2 |
| 14 | 1 | 453.59 | 2 |

Tabla H.4 Valores para una alimentación Líquida

| Valores para una Alimentación Líquida del Animal | | |
|---------------------------------------------------------|-----------------|---------------------|
| Valor (Python) | Valor en Litros | Valor en mililitros |
| 17.5 | 0.25 | 250 |
| 35 | 0.5 | 500 |
| 52.5 | 0.75 | 750 |
| 70 | 1 | 1000 |

Cambio de valores en el programa en la placa Arduino uno para la alimentación del animal.

Vamos a la barra de inicio del escritorio del Raspberry Pi para identificar el ítem Electrónica y seguido Arduino IDE y se abre el programa de Arduino.



Anexo H.6 Inicio del programa Arduino Uno en Raspberry Pi.

Ahora nos dirigimos a la barra de herramientas de Arduino a FILE y Abrimos la carpeta que contiene el programa de alimentación doméstica.

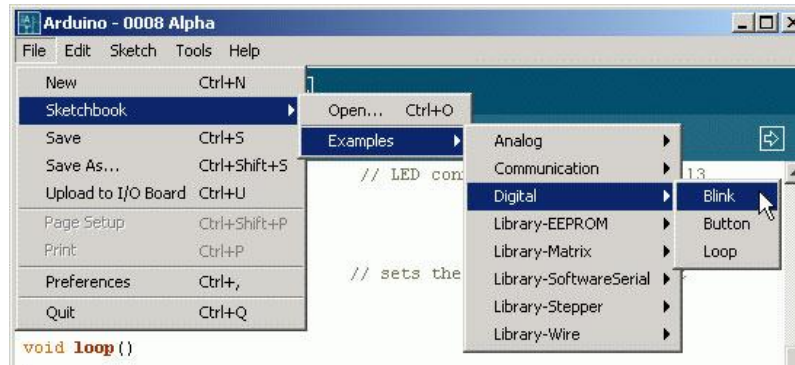


Figura H.7 Selección de carpeta que contiene el programa de Alimentación de animales.

Abrimos el programa y nos dirigimos a la línea de //Alimentación de sólidos a `if(val>7)` y cambiamos el valor deseado según la tabla de alimentación de animales.

```

Alimentacioncandatos
    if(in == 'I'){
        aux1 = (100 - valor1);
        Serial.println(aux1);
    }
}
{//alimentacion de solidos
val = (5.0 * analogRead(0)*100.0)/1023.0;
if (val > 7){
    digitalWrite(12,LOW);
}else
    digitalWrite(12,HIGH);
}

```

Figura H.8 Programación en Arduino Uno de la Alimentación Sólida.

Seguidamente cambiamos los valores de la alimentación líquida en la línea que dice Alimentación Líquida en `if(val1>44)` y cambiamos el valor deseado según la tabla de alimentación del animal.

```

}
{//alimentacion de liquido
val1 = (5.0 * analogRead(1)*100.0)/1023.0;

if (val1 > 44){
    digitalWrite(11,LOW);
}else
    digitalWrite(11,HIGH);
}

```

Figura H.9 Programación en Arduino Uno de la Alimentación Sólida.

Después de realizar los cambios de valores realizamos la verificación del programa en busca de errores.



Figura H.10 Panel del Arduino Uno, verificar el programa en busca de errores.

Luego guardamos los cambios realizados en el programa.



Figura H.11 Panel del Arduino Uno, Guardar el programa.

Y cargamos el programa en la tarjeta Arduino y posterior a esto estamos listos para proceder con la alimentación del can.

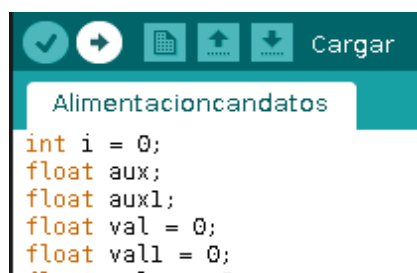


Figura H.12 Panel del Arduino Uno, cargar el programa.

Ahora estamos listos para trabajar con nuestro proyecto y aprovechar al máximo de su proceso de alimentación y vigilancia de su mascota apreciada.

ANEXO I
Instalación y Configuración: Xming

Instalación y Configuración: Xming

Instalación.

1. Abrimos la carpeta en donde hemos descargado Xming y hacemos doble clic sobre su icono. En nuestro caso, lo hemos descargado en el Escritorio de Windows.



Figura I.1 Icono del Xming en el escritorio de Windows

2. Hacemos toda la instalación por defecto:



Figura I.2 Inicio de la instalación del programa.



Figura I.3 Dirección donde se instalará el programa Xming.

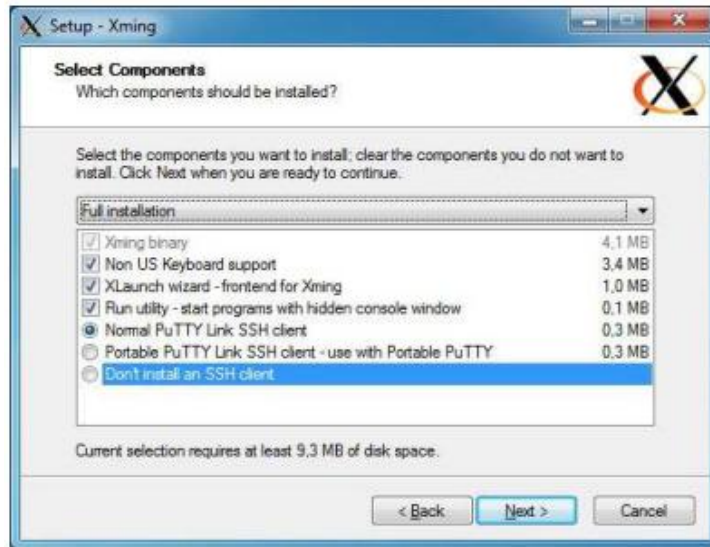


Figura I.4 Configuraciones del Xming

3. Nótese que la instalación por defecto es la “Full instalation” y que tiene marcado el checkbox de “Run utility – start programs with hidden console window” (ver captura de pantalla de arriba).



Figura I.5 Crea el programa con el nombre Xming en la barra de inicio.



Figura I.6 Ventana de instalación del Xming.



Figura I.7 Finalización de la instalación de Xming.

4. Como no hemos creado ningún acceso directo para Xming, buscamos la carpeta creada en la instalación (que debiera ser C:\Program Files\Xming) y buscamos XLaunch y configuramos la apariencia de la ventana que se abrirá en nuestro equipo al lanzar una aplicación gráfica remota. Podemos dejar los valores por defecto o cambiar alguno de ellos si nos es más cómodo.

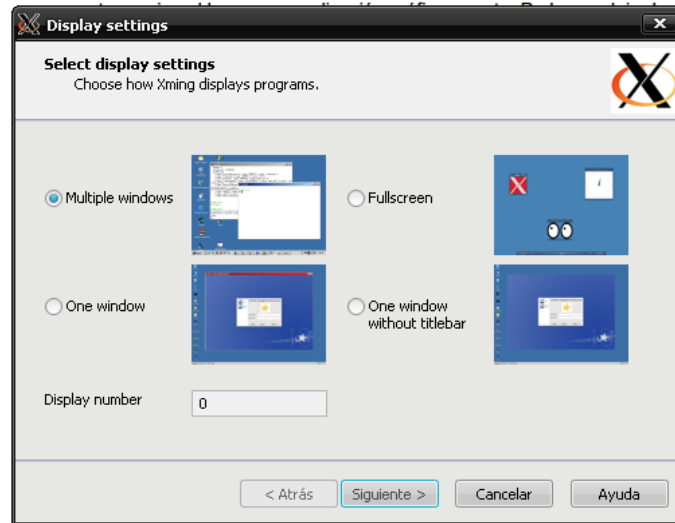


Figura I.8 Configuración del Xlaunch.

5. Ahora seleccionamos la segunda opción de start a program.

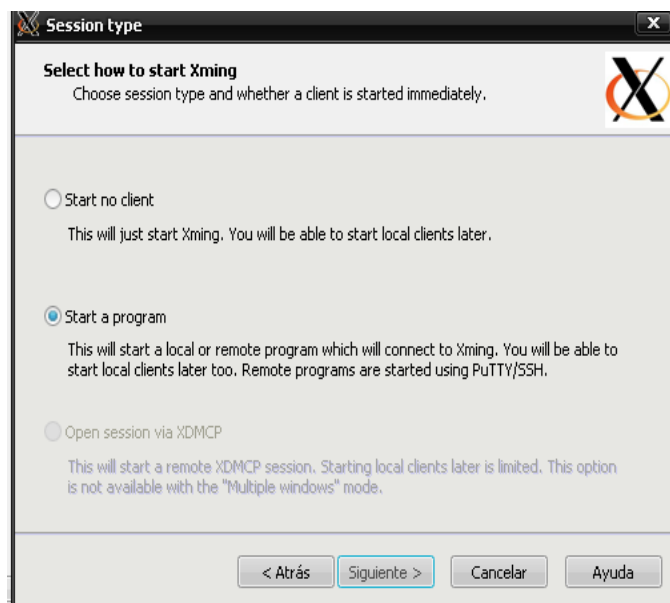


Figura I.9 Ventana para colocar una dirección directa al Raspberry pi.

6. En la siguiente ventana colocamos en Start program “startlxde” y en Run Remote la dirección del Raspberry Pi que es 192.168.1.109 y posterior el nombre y una contraseña la cual será ingresada cuando vamos a iniciar el programa.

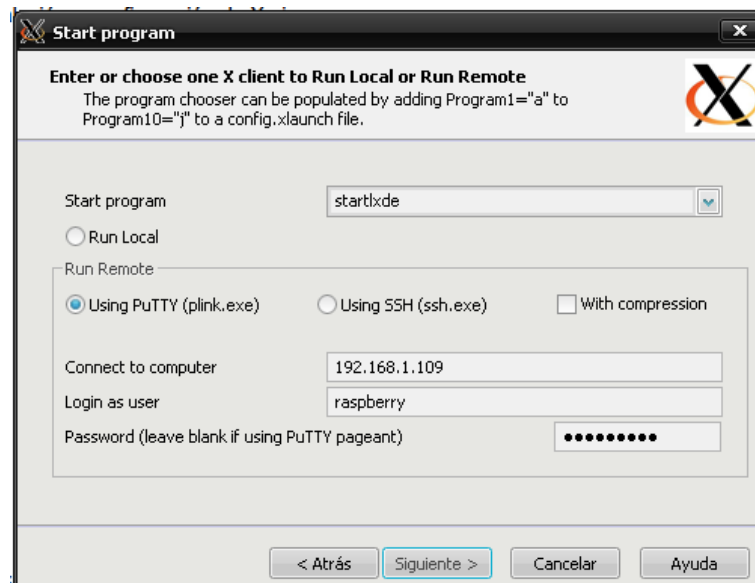


Figura I.10 Direccionamiento y contraseña para acceder al Raspberry pi.

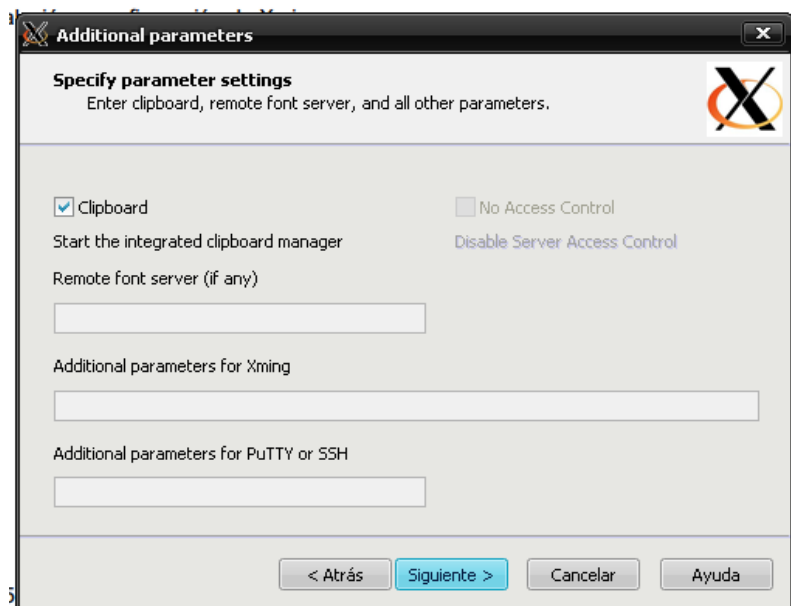


Figura I.11 Ventana donde dejamos como esta por defecto y continuamos.

7. En el último paso, podemos guardar la configuración que hemos realizado para que sea la configuración por defecto (si queremos modificarla, sólo tenemos que volver a lanzar el XLaunch y repetir el proceso). Sobrescribimos el archivo de configuración.

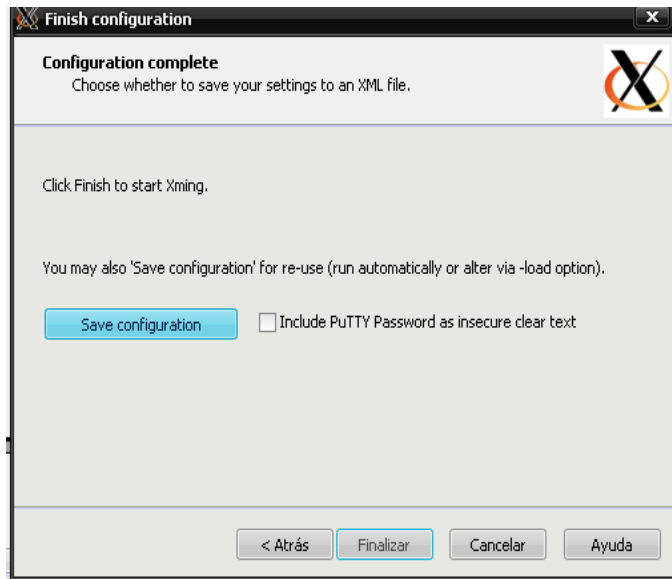


Figura I.12 Ventana donde guardamos la configuración del Xlaunch.

8. Después de guardar pulsamos finalizar y terminamos con la instalación y configuración.

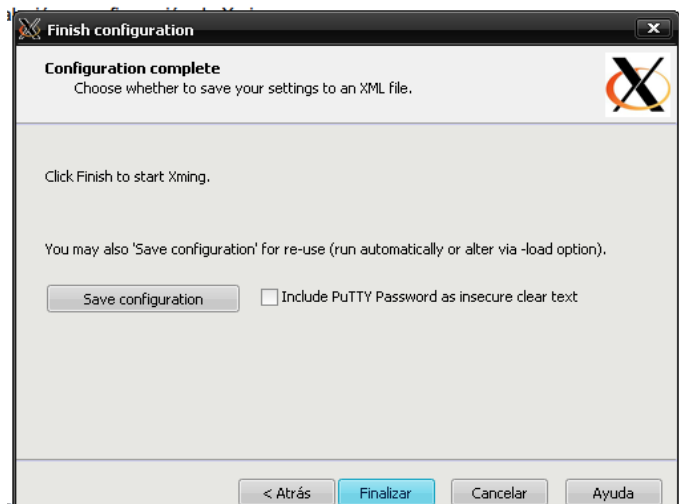


Figura I.13 Finalizamos la configuración del xlaunch.

9. Cuando vayamos a ingresar al Raspberry pi damos doble clic en nuestro icono de acceso ya previamente configurado, la cual nos saldrá una ventana donde tendremos que introducir la contraseña que anteriormente introducimos y se realizara la conexión con el escritorio del Raspberry pi.

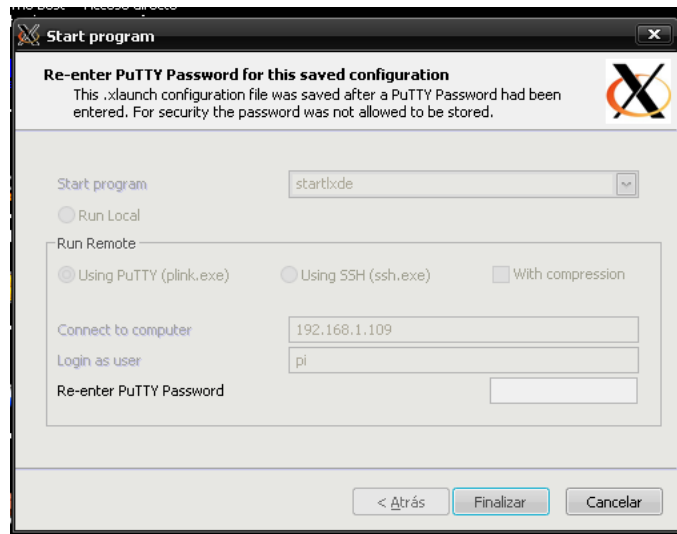


Figura I.14 Ventana de acceso al escritorio del Raspberry pi al colocar la contraseña establecida.

GLOSARIO DE TERMINOS

API Interfaz de Programación de Aplicaciones

CC Corriente Continua

CA Corriente Alterna

CI Circuito Integrado

FTP Par Trenzado con Blindaje Global

GND Referencia a Tierra

HTML Lenguaje de Marcado Hipertextual

HTML5 Lenguaje de Marcado Hipertextual 5

HTTP Protocolo de Transferencia de Hipertexto

ICSP Programación Serial en Circuito

IDE Entorno de Desarrollo Integrado

IP Protocolo de Internet

PWM Modulación por Ancho de Pulsos

SCK Señal de Reloj

STP Par Trenzado Apantallado

SPI Interface Periférica Serie

SSC Sistema de Seguridad y Confort

STP Cable de Par Trenzado Apantallado

TCP Protocolo de Control de Transmisión

UART Transmisor-Receptor Asíncrono Universal

UDP Protocolo de Datagrama de Usuario

UPS Sistemas de Energía Ininterrumpida

URL Localizador de Recursos Uniforme

USB Bus de Serie Universal

UTP Par Trenzado no Blindado

V Voltios

PC Computadoras

LAN (Local Area Network): Son las redes de área local

WAN (Wide Area Network): Son redes que cubren un espacio muy amplio, conectando a ordenadores de una ciudad o un país completo

HMI (Human Machine Interface): Se usa para referirse a la interacción entre humanos y máquinas.

Periférico Es un dispositivo electrónico físico que se conecta o acopla a una computadora, pero no forma parte del núcleo básico (CPU, memoria, placa madre, alimentación eléctrica) de la misma.

ROM Memoria no volátil de sólo lectura.

VCR Grabador de Video

DVR Grabador de Video Digital

Web cam Cámara Web o Cámara de Red

Microcontrolador: Es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: unidad central de procesamiento, memoria y unidades de E/S (entrada/salida).

ISO (Organización Internacional de Normalización)

TCP/IP TCP (Transmission Control Protocol) y el IP (Internet Protocol),

MODEM modulador y demodulador de datos

OSI Sistemas de interconexión abierta.

ADSL Línea de abonado digital asimétrica.

GPRS Servicio de comunicación basada en paquetes de datos

TX Transmisión

RX Recepción

Latacunga, Noviembre del 2013.

AUTORÍA

ELABORADO POR:

Christian Geovanny Masaquiza Criollo
CI: 1803754223

Luis Orlando Saquina Sailema
CI: 1804025094

APROBADO POR:

Ing. José Bucheli.
**DIRECTOR DE LA CARRERA DE INGENIERÍA EN ELECTRÓNICA E
INSTRUMENTACIÓN**

CERTIFICADO POR:

Dr. Rodrigo Vaca
**SECRETARIO ACADÉMICO
UNIDAD DE ADMISIÓN Y REGISTRO**