



**UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN**

**“SISTEMA DE TELEMETRÍA DE LA PRODUCCIÓN DE LECHE Y
CONTROL DEL ORDEÑO MECÁNICO DEL GANADO BOVINO,
BASADO EN PLATAFORMAS DE HARDWARE Y SOFTWARE
LIBRE, EN LA GRANJA EXPERIMENTAL LA PRADERA DE LA
UNIVERSIDAD TÉCNICA DEL NORTE.”**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERÍA EN ELECTRÓNICA Y REDES DE
COMUNICACIÓN**

AUTOR: OLAVO GABRIEL PAVÓN IPIALES

DIRECTOR: ING. OMAR OÑA

IBARRA-ECUADOR

2016



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital Institucional determina la necesidad de disponer de textos completos en formato digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información.

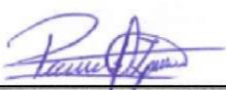
DATOS DEL CONTACTO	
Cédula de Identidad	1003224753
Apellidos y Nombres	Pavón IpiALES Olavo Gabriel
Dirección	José Nicolás Vacas 3-70 y Juan J. Flores
Email	gabrielcime@live.com
Teléfono Fijo	062955325
Teléfono Móvil	0990591457

DATOS DE LA OBRA	
Título	SISTEMA DE TELEMETRÍA DE LA PRODUCCIÓN DE LECHE Y CONTROL DEL ORDEÑO MECÁNICO DEL GANADO BOVINO, BASADO EN PLATAFORMAS DE HARDWARE Y SOFTWARE LIBRE, EN LA GRANJA EXPERIMENTAL LA PRADERA DE LA UNIVERSIDAD TÉCNICA DEL NORTE
Autor	Pavón IpiALES Olavo Gabriel
Fecha	01/03/2016
Programa	Pregrado

Título por el que se Aspira	Ingeniero en Electrónica y Redes de Comunicación
Director	Ing. Omar Oña

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Olavo Gabriel Pavón Ipiales, con cédula de identidad Nro. 1003224753, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad de material y como apoyo a la educación, investigación y extensión, en concordancia con la ley de Educación Superior Artículo 144.



Firma

Nombre: Olavo Gabriel Pavón Ipiales

Cedula: 1003224753

Ibarra a los 01 días del mes de Marzo del 2016



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

Yo, OLAVO GABRIEL PAVÓN IPIALES, con cédula de identidad Nro.100322475-3, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, artículo 4,5 y 6, en calidad de autor del trabajo de grado denominado: **“SISTEMA DE TELEMETRÍA DE LA PRODUCCIÓN DE LECHE Y CONTROL DEL ORDEÑO MECÁNICO DEL GANADO BOVINO, BASADO EN PLATAFORMAS DE HARDWARE Y SOFTWARE LIBRE, EN LA GRANJA EXPERIMENTAL LA PRADERA DE LA UNIVERSIDAD TÉCNICA DEL NORTE”**, que ha sido desarrollado para optar por el título de Ingeniero en Electrónica y Redes de Comunicación, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes mencionada, aclarando aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional.

En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica del Norte.

Firma

Nombre: Olavo Gabriel Pavón Ipiales

Cedula: 1003224753

Ibarra a los 01 días del mes de Marzo del 2016

UNIVERSIDAD TÉCNICA DEL NORTE



CONSTANCIA

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrollo, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

En la ciudad de Ibarra, Marzo de 2016

El Autor:

Olavo Gabriel Pavón Ipiales

C.I.: 1003224753



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DECLARACIÓN

Yo, Olavo Gabriel Pavón IpiALES, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; y que éste no ha sido previamente presentado para ningún grado o calificación profesional.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Universidad Técnica del Norte, según lo establecido por las Leyes de Propiedad Intelectual, Reglamentos y Normatividad vigente de la Universidad Técnica del Norte

Olavo Gabriel Pavón IpiALES

C.I.: 1003224753

UNIVERSIDAD TÉCNICA DEL NORTE



CERTIFICACIÓN

INGENIERO OMAR OÑA, DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN.

CERTIFICA:

Que, el presente Trabajo de Titulación “**SISTEMA DE TELEMETRÍA DE LA PRODUCCIÓN DE LECHE Y CONTROL DEL ORDEÑO MECÁNICO DEL GANADO BOVINO, BASADO EN PLATAFORMAS DE HARDWARE Y SOFTWARE LIBRE, EN LA GRANJA EXPERIMENTAL LA PRADERA DE LA UNIVERSIDAD TÉCNICA DEL NORTE**” fue desarrollado en su totalidad por el Sr. Olavo Gabriel Pavón Ipiales portadora de la cédula de identidad número: 1003224753, bajo mi supervisión.

Ing. Omar Oña

DIRECTOR DE PROYECTO

AGRADECIMIENTO

Agradezco a Dios por haberme dado la fuerza, sabiduría y fortaleza para poder llegar a culminar mis estudios; a mi madre, abuelos y familiares, por brindarme siempre su incondicional apoyo en cada momento de mi vida y ser un pilar fundamental para llegar a culminar cada una de mis metas.

A mi hija Bryhana, Cristina y Matías, quienes son mi familia y han permanecido a mi lado durante este proceso y todo proceso de mi vida, brindándome palabras de aliento y amor.

A todos los docentes de la facultad por sus aportes y por guiarme en mi formación académica y personal.

Al Ing. Henry Cervantes, director de Tesis, al Ing. Jaime Michilena docente de la materia Trabajo de Grado e Ing. Carlos Vásquez por sus aportes y asesorías invaluable para la elaboración de éste proyecto.

A los amigos, que con su apoyo moral, alentaron al desarrollo de este trabajo, en especial a la familia Lara.

A todos y cada uno de ustedes, nuevamente.

Gracias.

Olavo Gabriel Pavón Ipiales.

DEDICATORIA

Dedico este proyecto a mi madre Elizabeth, mi hija Bryhana, mi abuelita Cecilia y mi mujer Cristina, que con su incondicional amor y ejemplos dignos de superación y entrega, dentro de mi formación académica, así como integral, han logrado que este esfuerzo se culmine con éxito.

A mis familiares y amigos quienes han estado siempre a mi lado brindándome su ayuda con el fin de que logre culminar esta meta.

Olavo Gabriel Pavón Ipiales.

CONTENIDO

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE.....	I
CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	III
CONSTANCIA	IV
DECLARACIÓN	V
CERTIFICACIÓN.....	VI
AGRADECIMIENTO.....	VII
DEDICATORIA.....	VIII
ÍNDICE DE FIGURAS	XV
ÍNDICE DE TABLAS.....	XVIII
RESUMEN.....	XIX
ABSTRACT	XXI
CAPÍTULO I.....	1
1.1 PROBLEMA	1
1.2 OBJETIVOS.....	2
1.2.1 Objetivo General.....	2
1.2.2 Objetivos Específicos	2
1.3 ALCANCE	3
1.4 JUSTIFICACIÓN.....	6
CAPITULO II.....	8
2.1 SISTEMA ANÁLOGOS Y DIGITALES	8
2.1.1 Señal Analógica	9
2.1.2 Señal Digital	9
2.1.3 Sistema General	10
2.1.4 Sistema Analógico	13

2.1.5	Sistema Digital.....	13
2.1.6	Sistema Analógico-Digital.....	14
2.1.7	Ventajas de los sistemas digitales respecto a los analógicos	15
2.2	SENSORES	16
2.2.1	Características de operación	17
2.2.2	Factores a Considerar.....	18
2.2.3	Clasificación de sensores	20
2.2.4	Tipos de sensores	21
2.2.5	Sensores primarios	22
2.3	ACTUADORES	26
2.3.1	Actuadores Neumáticos	26
2.3.2	Actuadores Hidráulicos.....	27
2.3.3	Actuadores eléctricos	28
2.3.4	Características importantes a considerar.....	29
2.4	ÁREAS DE APLICACIÓN DE LOS SENSORES Y ACTUADORES.....	30
2.5	ESTUDIO DE LOS DIFERENTES DISPOSITIVOS DE PROPÓSITO GENERAL.	31
2.5.1	Arduino.	31
2.5.2	Alternativas del microcontrolador Arduino	33
2.5.3	Tabla comparativa de los Arduinos mencionados	39
2.5.4	MINICOMPUTADORA	39
2.5.5	Tabla comparativa de las minicomputadoras mencionadas.....	46
2.6	PROTOCOLOS DE ACCESO A LA RED.....	47
2.6.1	Protocolos y comunicaciones de red.....	47
2.6.2	Medios para acceder a la red.....	48
2.6.3	Propiedades de los medios inalámbricos	49
2.6.4	LAN inalámbrica	51

2.6.5	Ethernet	55
2.7	ESTUDIO DE LA TECNOLOGÍA RFID.....	57
2.7.1	Introducción	57
2.7.2	Componentes del Sistema	58
2.7.3	FRECUENCIAS DE OPERACIÓN	65
2.8	GENERALIDADES DE LOS SERVIDORES	67
2.8.1	Modelo cliente-servidor	67
2.8.2	Protocolos de capa aplicación.....	68
CAPÍTULO III		76
3.1	CONSIDERACIONES DEL DISEÑO	76
3.2	DESCRIPCIÓN DEL FUNCIONAMIENTO DEL SISTEMA	77
3.2.1	Primera Etapa.....	77
3.2.1.1	Recursos.....	77
3.2.1.2	Acción	78
3.2.2	Segunda Etapa.....	78
3.2.2.1	Recursos.....	79
3.2.2.2	Acción	79
3.2.3	Tercera Etapa	80
3.2.3.1	Recursos.....	80
3.2.3.2	Acción	81
3.2.4	Cuarta Etapa.....	82
3.2.4.1	Recursos.....	82
3.2.4.2	Acción	83
3.2.5	Quinta Etapa	84
3.2.5.1	Recursos.....	84
3.2.5.2	Acción	84
3.2.6	Sexta Etapa	85

3.2.6.1	Recursos	85
3.2.6.2	Acción	85
3.3	REQUERIMIENTOS TECNOLÓGICOS (COMPONENTES ELECTRÓNICOS Y RECURSOS UTILIZADOS)	86
3.3.1	ARDUINO MEGA 2560	86
3.3.1.1	Recursos utilizados de Arduino Mega 2560	87
3.3.2	Raspberry Pi B+.....	88
3.3.2.1	Recursos utilizados de la mini computadora Raspberry Pi B+.....	88
3.3.3	MODULO RFID RC522	89
3.3.3.1	Características del módulo RFID RC522	89
3.3.4	SENSOR DE FLUJO FS300A	90
3.3.4.1	Características del sensor	91
3.3.4.2	Especificaciones del sensor.....	91
3.3.5	VÁLVULA SOLENOIDE.....	92
3.3.5.1	Características de la Válvula solenoide	92
3.3.5.2	Especificaciones de la Válvula solenoide	92
3.3.6	PANTALLA GRAFICA GLCD 128X64	93
3.3.6.1	Características de una GLCD 128x64	94
3.3.6.2	Descripción de los pines de conexión de una GLCD 128x64	95
3.3.7	MOTOR ELÉCTRICO	96
3.3.8	MODULO RTC (REAL TIME CLOCK) 3231	97
3.3.9	SERVIDOR LAMP	98
3.4	DISEÑO DEL HARDWARE DEL SISTEMA.....	98
3.4.1	Arquitectura del Sistema.....	98
3.4.2	Diagramas de bloques	99
3.4.3	Diagrama de conexión (esquemático).....	99
3.4.4	Distribución de pines del módulo Arduino Mega 2560.....	105

3.5	DISEÑO DEL SOFTWARE DEL SISTEMA	107
3.5.1	Diagramas de flujo.....	107
3.5.2	Algoritmo de programación.....	111
3.6	DISEÑO DE PLACAS DE CIRCUITOS IMPRESOS	112
3.6.1	Esquemático de conexión de la placa principal	112
3.6.2	Placa principal (componentes electrónicos)	113
3.6.3	Placa secundaria.....	114
3.7	Montaje final en Protoboard	115
	CAPÍTULO IV	116
4.1	IMPLEMENTACIÓN DEL SISTEMA	116
4.1.1	Montaje del Gabinete.....	116
4.2	PRUEBAS DEL FUNCIONAMIENTO DEL SISTEMA.	118
4.2.1	Pruebas de hardware.	118
4.2.2	Pruebas de Software.....	124
4.2.2.1	Pruebas realizadas en el microcontrolador Arduino Mega 2560	124
4.2.2.2	Pruebas realizadas en la minicomputadora Raspberry Pi B+ para la gestión de Base de Datos	126
4.2.2.3	Resultados de almacenamiento de los datos en un archivo de texto en la minicomputadora	130
4.2.2.4	Resultados de acceso y de transferencia de la información a través de un servidor FTP	131
4.2.2.5	Verificación del servidor web para monitorear y consultar los resultados previos del ganado bovino.....	134
4.2.2.6	Verificación del envío del archivo con la información del bovino a través de un correo electrónico.	136
4.3	ANÁLISIS DE RESULTADOS.....	137
	CAPÍTULO V	139
5.1	COSTOS DEL SISTEMA.....	139

5.1.1	Costos del diseño y elaboración del prototipo.....	139
5.2	CONCLUSIONES.....	142
5.3	RECOMENDACIONES	144
	REFERENCIAS BIBLIOGRÁFICAS:	146
	ANEXO 01: FIRMWARE DEL MICROCONTROLADOR ARDUINO MEGA 2560 ..	149
	ANEXO 02: FIRMWARE DEL SCRIPT EN RASPBERRY PI B+ MEDIANTE PYTHON.....	162
	ANEXO 03: FIRMWARE DEL SCRIPT EN RASPBERRY PI B+ PARA REVISAR INFORMACIÓN EN PAGINA WEB MEDIANTE PHP Y HTML	167
	ANEXO 04: CÁLCULOS DE LOS RESULTADOS OBTENIDOS DEL SENSOR DE FLUJO PARA VER LA CANTIDAD DE LECHE PRODUCIDA EN LITROS DE UN BOVINO.....	173
	ANEXO 05: IMÁGENES	174
	MANUALES	179

ÍNDICE DE FIGURAS

Figura 1 Representación de una magnitud analógica	9
Figura 2 Representación de una magnitud digital	10
Figura 3 Ejemplo de un sistema	11
Figura 4 Sistema General	12
Figura 5 Ejemplo de sistema analógico	13
Figura 6 Ejemplo de sistema digital	14
Figura 7 Ejemplo de sistema analógico-digital	15
Figura 8 Efecto Hall	25
Figura 9 Sensores de efecto Hall: a) lineal; b) de umbral	25
Figura 10 Logo de Arduino	31
Figura 11 Parte frontal y trasera de Arduino Uno	33
Figura 12 Parte frontal y trasera de Arduino Leonardo.....	34
Figura 13 Parte frontal y trasera de Arduino Mega 2560.....	36
Figura 14 Parte frontal y trasera de Arduino Due.	37
Figura 15 Parte frontal y trasera de Intel Galileo	40
Figura 16 Parte frontal de Raspberry PI	42
Figura 17 Parte frontal y trasera de PcDuino V1.....	44
Figura 18 Modelos de Red	48
Figura 19 Ejemplos de señalización para medios inalámbricos, de cobre y de fibra óptica	49
Figura 20 Diferencias entre los medios inalámbricos	51
Figura 21 Trama Protocolo LAN inalámbrico 802.11	55
Figura 22 Trama Ethernet.....	57
Figura 23 Esquema general de funcionamiento de un sistema RFID.....	58
Figura 24 Arquitectura Básica tecnología RFID	58
Figura 25 Ejemplos de tags de diferentes formas y tamaños	59
Figura 26 Estructura Básica de un Tag RFID.....	60
Figura 27 Polarización Lineal.....	63
Figura 28 Polarización Circular.....	64
Figura 29 Lector fijo seteado para leer etiquetas automáticamente	65
Figura 30 Modelo cliente-servidor	68
Figura 31 Protocolos de capa Aplicación del modelo OSI y TCP/IP.....	68

Figura 32 Proceso FTP	70
Figura 33 Programas de emulación de terminal	71
Figura 34 Funcionamiento de las tecnologías ricas de Internet usando tecnologías en el lado del cliente.....	72
Figura 35 Funcionamiento de las aplicaciones creadas para ser interpretadas en el lado del servidor.....	73
Figura 36 Servidor LAMP.....	74
Figura 37 Arduino Mega 2560	86
Figura 38 Raspberry Pi B+	88
Figura 39 Módulo RFID RC522.....	89
Figura 40 Sensor de flujo FS300A	90
Figura 41 Válvula Solenoide	92
Figura 42 Parte frontal de pantalla grafica GLCD 128x64	93
Figura 43 Pines de la GLCD 128X64.....	96
Figura 44 Motor de plumas de camión Hyundai HD72	96
Figura 45 Modulo RTC 3231	97
Figura 46 Arquitectura del sistema.....	98
Figura 47 Diagramas de bloques del sistema	99
Figura 48 Diagrama de conexión entre Arduino Mega 2560 y pantalla grafica GLCD 128x64	100
Figura 49 Diagrama de conexión entre Arduino Mega 2560 y lector RFID RC522.....	101
Figura 50 Diagrama de conexión entre Arduino Mega 2560 y Raspberry Pi B+	101
Figura 51 Diagrama de conexión del Arduino Mega 2560 y los demás sistemas.....	102
Figura 52 Diagrama de conexión del módulo RTC y Raspberry Pi B+	103
Figura 53 Diagrama de conexión del pulsador y Raspberry Pi B+	104
Figura 54 Diagrama de conexión del Sistema completo	104
Figura 55 Diagrama de flujo del sistema.....	110
Figura 56 Esquemático de conexión del sistema.....	113
Figura 57 Diseño de la placa principal	114
Figura 58 Diseño de la placa secundaria	115
Figura 59 Pruebas previas del sistema en tablero de ensayo electrónico	115
Figura 60 Parte interna del gabinete eléctrico	117
Figura 61 Parte externa del gabinete eléctrico.....	118
Figura 62 Mensaje de bienvenida en GLCD	119

Figura 63 Mensaje de “Identificación correcta” del bovino en GLCD	120
Figura 64 Led de visualización de pulsador activo	120
Figura 65 Tiempo de espera en GLCD para colocar pezoneras	121
Figura 66 Luz piloto activado para inicio de proceso de extracción de leche.....	121
Figura 67 Leds de visualización de inicio de proceso y válvula solenoide activados.....	122
Figura 68 Proceso de medición de datos en funcionamiento	122
Figura 69 Luz piloto activado para indicar fin de proceso de extracción de leche	123
Figura 70 Leds de visualización de fin de proceso y motor eléctrico activados	123
Figura 71 Mensaje de “Identificación correcta” y “Activación de motor” en monitor serial	124
Figura 72 Mensajes del proceso de medición de datos en funcionamiento en monitor serial	125
Figura 73 Fin del proceso de extracción y envío de datos a la minicomputadora.....	126
Figura 74 Ejecución manual del script del sistema	127
Figura 75 Impresión en consola de los datos obtenidos del microcontrolador	127
Figura 76 Gestión de base de datos por interfaz web	128
Figura 77 Datos almacenados en la base de datos.....	128
Figura 78 Exportación de los datos	129
Figura 79 Datos en formato pdf.....	129
Figura 80 Datos en formato Word.....	130
Figura 81 Ejecución del archivo de texto	130
Figura 82 Verificación de datos almacenados en archivo de texto	131
Figura 83 Conexión al servidor FTP	132
Figura 84 Selección de archivo que se desea transferir.....	133
Figura 85 Transferencia correcta de archivo	133
Figura 86 Verificación de archivo en sitio local.....	134
Figura 87 Autenticación en el servidor Web	135
Figura 88 Consultar resultados de cada bovino en servidor Web	135
Figura 89 Verificación de los resultados del bovino seleccionado en un navegador Web	136
Figura 90 Verificación de envío del archivo por correo electrónico.....	137

ÍNDICE DE TABLAS

Tabla 1 Factores a considerar en la elección de un sensor	19
Tabla 2 Resumen de las características técnicas de Arduino	34
Tabla 3 Resumen de las características técnicas de Arduino Leonardo	35
Tabla 4 Resumen de las características Técnicas de Arduino Mega.....	36
Tabla 5 Resumen de las características Técnicas de Arduino Due	38
Tabla 6 Comparativa entre Arduinos.....	39
Tabla 7 Comparativa de las minicomputadoras	46
Tabla 8 Diferencias entre estándares de WI-FI 802.11	53
Tabla 9 Diferencias técnicas entre tecnologías RFID activas y pasivas.....	60
Tabla 10 Capacidades funcionales de tecnologías RFID activos y pasivos	61
Tabla 11 Comparativa entre características de tags pasivos y tags activos.....	62
Tabla 12 Bandas de frecuencias utilizadas en RFID	66
Tabla 13 Características principales y ejemplos de aplicación de las distintas bandas de frecuencia utilizadas en RFID	66
Tabla 14 Servidores FTP más comunes	70
Tabla 15 Distribución de pines del microcontrolador Arduino Mega 2560 con los demás dispositivos	105
Tabla 16 Voltaje de funcionamiento de los componentes electrónicos utilizados	114
Tabla 17 Resultados obtenidos del sensor de flujo	138
Tabla 18 Costo del diseño y elaboración del sistema.....	139

RESUMEN

El presente proyecto propone el diseño y construcción de un sistema de telemetría que permita registrar la producción de leche del bovino, y un control automático del sistema de ordeño para evitar el sobre ordeño y la mastitis. Se basa en la tecnología RFID para la identificación del animal, y su verificación en una base de datos remota conectada a través de una red de área local. Además, se implementa un servidor Web que permita gestionar la base de datos, y un servidor FTP para transferir la información del ganado bovino. El sistema deberá enviar un correo electrónico al administrador o persona encargada de la granja con el contenido de la información del bovino.

El desarrollo del proyecto se encuentra distribuido en cinco capítulos que contienen la información concerniente al diseño, construcción e implementación del mismo.

El primer capítulo contiene los problemas, el objetivo general, los objetivos específicos, el alcance y su respectiva justificación del proyecto de titulación.

En el segundo capítulo se describen los fundamentos teóricos útiles para el desarrollo del proyecto, en este se detallan los aspectos más importantes de los dispositivos de propósito general, sensores, actuadores, la tecnología RFID, las generalidades de los servidores y los protocolos, y comunicaciones de red.

En el capítulo tres se procede con el diseño del sistema, tomando en cuenta los requerimientos de hardware y software que se necesita para la implementación adecuada del control y registro del sistema.

En el cuarto capítulo, se describe el proceso de implementación del sistema, la instalación, las pruebas respectivas, la depuración y corrección de errores tanto de software como de hardware.

Finalmente, el capítulo quinto se detalla un presupuesto referencial que abarca todos los componentes empleados en el desarrollo del sistema. También, se encuentran las conclusiones y recomendaciones obtenidas en el desarrollo del presente trabajo de titulación.

ABSTRACT

This project has as propose the designing and construction of a telemetry system that allows to record bovine milk and an automatic milking system control to prevent over milking and mastitis. It is based on RFID technology for animal identification, and for its verification in a remote database connected through a local network. Also, will be implemented a web server to manage the database, and an FTP server to transfer cattle information. The system has to send an email to the form manager or the person in charge of the farm with bovine information.

The project is distributed in five chapters containing the information such as designing, construction and implementation.

The first chapter contains the problems, the general objective, specific objectives, short term and the respective justification for a project.

The second chapter describes useful theoretical basis for the development of the project, describing the most important aspects of the general purpose devices, sensors, actuators, RFID technology, and an overview of servers, and communications network.

In the third chapter it was proceed to design the system, taking into account the hardware and software requirements needed for the proper implementation of the monitoring and recording system.

In the fourth chapter, the process of implementation of the system is described, its testing, and correction of hardware and software errors.

Finally, the fifth chapter has the budget to cover all expenses involved in developing of the system. In addition, conclusions and recommendations resulting from the development of this graduation project.

CAPÍTULO I

En el presente capítulo se dará a conocer los problemas, el objetivo general, los objetivos específicos, el alcance y su respectiva justificación del proyecto de titulación.

1.1 PROBLEMA

El ordeño se basa en la extracción de la leche de la glándula mamaria se hace en dos formas básicas: la natural, que es realizada por el becerro y; la artificial, que es realizada por el hombre, ya sea en forma manual o en forma mecánica (AVILA, p.143). El ordeño manual tiene el problema de calidad higiénica de la leche que es inferior en comparación con el ordeño mecánico, puesto que la leche se expone al medio ambiente y a las manos del ordeñador (AVILA, p.151). Produciendo un sin número de enfermedades, una enfermedad muy común es la mastitis, compleja y costosa enfermedad de los bovinos de la granja.

Los módulos o bretes con ordeño mecánico son muy utilizados por las granjas, puesto que es una forma eficiente de extracción de leche, pero son todavía muy artesanales. El sistema de vacío de las pezoneras, es controlado mediante una llave de paso la cual se debe abrir al momento de empezar a ordeñar al animal y cerrar al terminar, la manipulación de esta llave también se realiza manualmente; cuando se ha efectuado la extracción de la leche de la vaca, esta se deposita en un recipiente pequeño cerca al brete, para su almacenamiento; por estas razones el proceso a pesar que cuenta con un ordeño mecánico es muy lento, lo cual disminuye el número de animales que se pueden ordeñar en un día, afectando directamente a la producción (PÉREZ, 2009). Además si no es controlado el módulo puede producir sobre ordeño en el animal.

La granja experimental La Pradera-UTN, ubicada en la parroquia de Chaltura, perteneciente al cantón Ibarra, posee una extensión de 32 hectáreas y está dedicada a la experimentación en el área agrícola y pecuaria (FICAYA). Actualmente la granja cuenta con una ordeñadora mecánica y no cuenta con un registro automático de la producción de leche obtenida de cada animal, por lo que el registro se lo realiza manualmente en una hoja de papel sin ningún tipo de control, esta acción se presta a muchos inconvenientes.

El diseño de un sistema que controle la extracción de leche del animal es una de las muchas áreas en la ganadería lechera que puede ser controlada, para evitar el sobre ordeño y enfermedades comunes como la mastitis en los bovinos de la granja, además este sistema debe registrar automáticamente la producción de leche del bovino. Sin duda alguna este sistema superará los inconvenientes presentados, teniendo un control fiable de la producción, mejorando su eficiencia y rendimiento del ordeño, y por ende la calidad de vida del animal.

1.2 OBJETIVOS

1.2.1 Objetivo General

- ❖ Diseñar e Implementar un sistema de telemetría de la producción de leche, en la Granja Experimental La Pradera – UTN, para control de producción.

1.2.2 Objetivos Específicos

- ❖ Complementar el sistema mecánico existente con dispositivos que permitan el control automático del sistema de ordeño con sensores y actuadores, para evitar el sobre ordeño del animal y consecuentemente la mastitis.

- ❖ Diseñar el algoritmo y código de programación de la identificación electrónica del animal basado en la tecnología RFID.
- ❖ Crear una base de datos para almacenar automáticamente la fecha, hora y datos obtenidos de la producción de leche diaria del animal de la granja.
- ❖ Habilitar un servidor web que permita monitorear y consultar los resultados previos del ganado bovino mediante el desarrollo de una interfaz gráfica.
- ❖ Generar un algoritmo de programación que permita el envío de un archivo con toda la información diaria del ganado bovino a través de un correo electrónico, con destino hacia el administrador de la granja.
- ❖ Crear un servidor FTP para que el administrador de la granja pueda acceder y transferir la información del ganado bovino en cualquier instante.
- ❖ Realizar la puesta en marcha y ejecución de pruebas del sistema para comprobar el funcionamiento.

1.3 ALCANCE

Este proyecto se desarrollará para el sector agropecuario, consiste en el diseño e implementación de un sistema de Telemetría de la producción de leche y control del ordeño del ganado bovino, en la Granja Experimental La Pradera de la Universidad Técnica del Norte. El desarrollo del presente proyecto se basa en dos plataformas de hardware y software libre: el microcontrolador Arduino y la mini PC.

Se realizará un proceso de selección para la mini PC entre las plataformas de software y hardware de código abierto como: Raspberry Pi, PcDuino o Intel Galileo.

La primera etapa, se basa en la identificación electrónica del animal a través de la tecnología inalámbrica del módulo RFID (Identificación por Radiofrecuencia), donde cada bovino tendrá incluido un dispositivo Tag RFID en su arete, y mediante el lector RFID transmitir la identidad o identificador de cada bovino al ingresar a la estación de ordeño. Este dato del identificador es enviado mediante el lector RFID hacia el microcontrolador. Para esta etapa es fundamental la presencia del operador en caso de que la etiqueta no sea identificada por el módulo lector RFID, consecuentemente el operador tiene que acercar la etiqueta al módulo para que pueda ser identificado correctamente el bovino.

En la segunda etapa, una vez obtenido el identificador del bovino, se procede a registrar automáticamente el nombre del animal, la fecha y hora de ingreso. El operador por medio de un pulsador o botón activará el sistema, y procederá a colocar las pezoneras en la ubre del animal, para luego por medio de sensores infrarrojos, ultrasónicos o potenciómetro lineal obtener la cantidad de leche producida. Este proceso se realizará a través del microcontrolador.

La tercera etapa, consta de un control del sistema, donde por medio de sensores de flujo, infrarrojo o ultrasónico, cuando el animal se haya ordeñado completamente, se interrumpirá su funcionamiento, y se retirará automáticamente las pezoneras del animal a través de un motor eléctrico, evitando el sobre ordeño del animal y consecuentemente la enfermedad de la mastitis. Al finalizar el proceso del ordeño, se activará una señal sonora y visual, para que el operario proceda a continuar con el ordeño del siguiente animal. Esta etapa también la realizará el microcontrolador.

La cuarta etapa, consiste en comunicar el microcontrolador con la mini PC, en la mini PC, se procederá a crear una base de datos para almacenar el nombre, fecha, hora y datos obtenidos de la producción de leche del animal de la plataforma Arduino.

Una vez creada la base de datos, se procederá habilitar un servidor Web que permita gestionar la base de datos obtenida, así como consultar resultados previos del ganado bovino a través de una interfaz gráfica. Se utilizará lenguajes de programación de script web como PHP, Python o HTML.

Como quinta etapa, se enviará un correo electrónico al administrador o persona encargada de la granja, que contenga un archivo con la información diaria de todo el ganado bovino. Esta etapa la realizará la mini PC y se utilizará como remitente el servicio de correo electrónico de Google (Gmail).

Como sexta etapa para que el administrador o persona encargada de la granja pueda acceder y transferir la información del ganado bovino, se creará un servidor FTP. Este proceso también se realizará en la mini PC.

Como tecnología de comunicación inalámbrica, se implementará la tecnología WiFi (Wireless Fidelity) para comunicarse con los servidores y la base de datos. Y así el administrador o encargado de la granja pueda gestionar los datos por medio de un Smartphone, Tablet, Laptop, etc.

Como última etapa, se realizarán las pruebas necesarias para que el sistema funcione correctamente, y las calibraciones necesarias de los componentes electrónicos. Y se presentará un análisis de los resultados obtenidos.

Se realizará un análisis de costo considerando las herramientas de hardware en la implementación del sistema.

1.4 JUSTIFICACIÓN

Este proyecto en el ámbito social ayuda a que la producción de leche del ganado bovino tenga un mayor grado de control e higiene, dando una pauta para poder implementar en diferentes granjas.

La Granja experimental La Pradera de la Universidad Técnica del Norte, necesita controlar el sistema mecánico de ordeño que posee para el ganado bovino, por tal razón, se implementará un sistema automático que permita el registro y control fiable de la producción de leche de cada animal. Además con este sistema se evitará las enfermedades comunes del ganado bovino, como es el sobre ordeño.

Con el avance acelerado de la tecnología y las nuevas tendencias de compartir información, se ha optado por desarrollar el presente proyecto con plataformas de hardware y software libre. Como mecanismos de comunicación inalámbrica, se optó por utilizar las tecnologías RFID y WIFI, debido a la facilidad que brindan para la implementación del

sistema y su bajo costo, además, que son tecnologías muy comunes que se pueden encontrar fácilmente en el mercado.

Con la realización de este proyecto, se podrá plasmar los conocimientos adquiridos en la carrera de Ingeniería, y a la vez, se aportará de una manera productiva a la Granja Experimental la Pradera, ya que con la aplicación de este proyecto, podrán optimizar el trabajo diario y obtener rentabilidad con la elaboración de productos lácteos.

CAPITULO II

En el presente capítulo se dará a conocer aspectos generales sobre los sistemas análogos y digitales, sensores, actuadores, y un estudio de los diferentes dispositivos de propósito general como el microcontrolador y la minicomputadora. Además se dará a conocer el modelo y referencia de las comunicaciones de red, los estándares inalámbricos y cableados que se utilizan en la actualidad para el desarrollo del siguiente proyecto de titulación.

También se conocerán aspectos fundamentales sobre la tecnología de identificación por Radiofrecuencia (RFID): componentes del sistema RFID, frecuencias de operación, ventajas, desventajas, usos y aplicaciones.

Y por último se dará a conocer generalidades de los servidores utilizados en este proyecto.

2.1 SISTEMA ANÁLOGOS Y DIGITALES

Los circuitos electrónicos según la naturaleza de los valores que toman las señales o magnitudes que intervienen en el sistema, se dividen en: analógicos y digitales. Donde la electrónica analógica utiliza magnitudes con valores continuos, mientras que la electrónica digital utiliza magnitudes con valores discretos. (ACHA).

2.1.1 Señal Analógica

Una señal analógica es aquella magnitud que puede tomar cualquiera de los infinitos valores del rango donde está definida, en cada instante de tiempo. Ejemplos de magnitudes analógicas que se presentan en la naturaleza: presión, humedad, temperatura, tensión eléctrica, etc. (ACHA).

En la Figura 1 se muestra un ejemplo de señal analógica. Se puede observar como la magnitud es la humedad relativa y varía de forma continua en un rango de valores a lo largo de un periodo de tiempo. (ACHA).

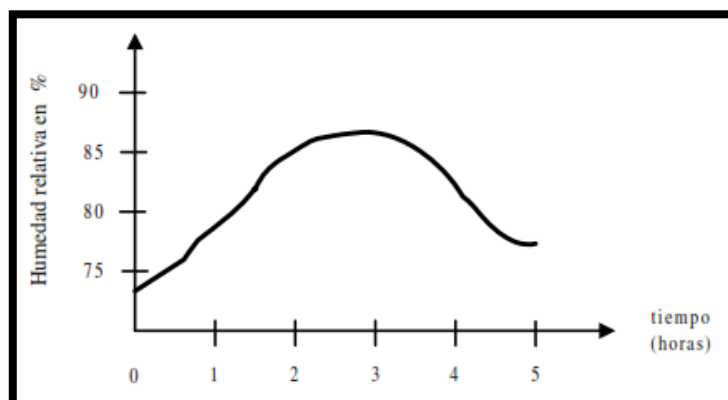


Figura 1 Representación de una magnitud analógica

Fuente: ACHA, S. C., Manuel; PEREZ, Julio; RIOSERAS, Miguel. Electrónica Digital Introducción a la Lógica Digital Teoría, Problemas y Simulación.: Alfaomega Ra-Ma

2.1.2 Señal Digital

Las señales digitales son aquellas cuya magnitud, pueden tomar un valor de entre un conjunto finito de valores discretos, en cada instante de tiempo. No existe valores intermedios por lo que se produce una discontinuidad entre el paso de un valor a otro. (ACHA).

La magnitud presenta dos estados: cerrado o abierto, alto (High) y bajo (Low), nivel de tensión alto (VH) o bajo (V), valor numérico 1 o 0, etc. A las señales digitales con dos estados se las denominan binarias, y constituyen la base de la electrónica digital. (ACHA).

En la Figura 2 se muestra un ejemplo de señal digital binaria, donde la magnitud representa el estado de un interruptor utilizado para activar/desactivar el foco de una habitación en función del tiempo. No existen valores intermedios entre el estado abierto y el estado cerrado. Por tanto, las señales digitales vienen caracterizadas por un número finito de posibles valores y las discontinuidades asociadas a dichos valores. (ACHA).

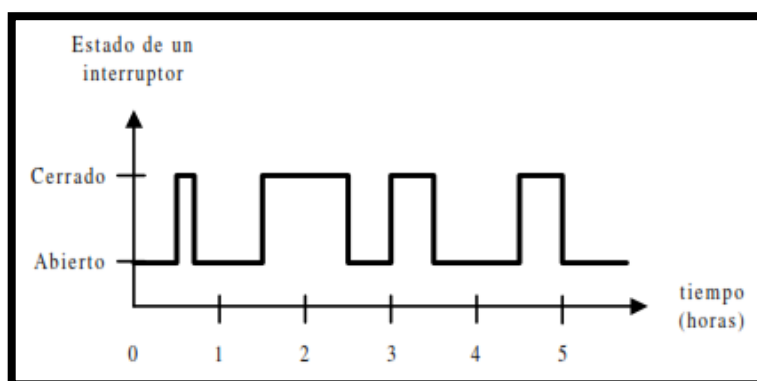


Figura 2 Representación de una magnitud digital

Fuente: ACHA, S. C., Manuel; PÉREZ, Julio; RIOSEAS, Miguel. *Electrónica Digital Introducción a la Lógica Digital Teoría, Problemas y Simulación.*: Alfaomega Ra-Ma

2.1.3 Sistema General

Un sistema es un conjunto de elementos con alguna característica en común. A los elementos de un sistema que así mismo tienen estructura de sistema, se les denomina subsistema. (ACHA).

El sistema más simple puede considerarse como una estructura cerrada con una entrada y una salida en donde el principal interés es conocer la relación entre estas dos variables. (STEREN).

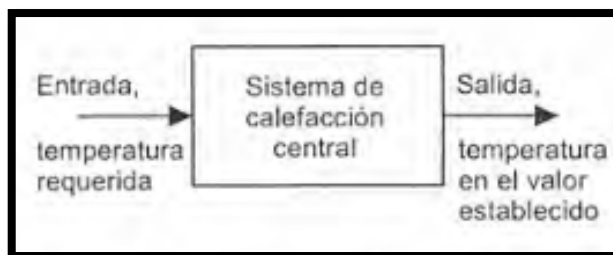


Figura 3 Ejemplo de un sistema

Fuente: BOLTON, W. (2006). Sistemas de Control Electrónico en la Ingeniería Mecánica y Eléctrica (Tercera ed.): Alfaomega

Un sistema sugiere un conjunto de elementos relacionados entre sí, para realizar una función bien definida que ninguno podría hacerlo por sí solo; un sistema basado en microprocesadores es un conjunto de componentes (circuitos electrónicos) interconectados capaces de efectuar sobre la información proveniente del proceso, representada en forma binaria, operaciones aritméticas y lógica, y realizar la transferencia de datos con los diferentes periféricos conectados al sistema. (COLLAGUAZO). Bajo este aspecto tenemos:

2.1.3.1 Periféricos de entrada (Sensores o captores)

Son aquellos que reciben la información de cambios físicos del proceso. Ejemplos: pulsadores, interruptores, contactos eléctricos, CAD, teclados, touch panel, etc. (COLLAGUAZO).

2.1.3.2 La Unidad de Control

Es la encargada de procesar la información recibida en base a un programa (secuencia lógica de instrucciones) y entrega los resultados para ser ejecutados. (COLLAGUAZO).

2.1.3.3 Periféricos de salida (Actuadores)

Son aquellos que ejecutan las órdenes en función de los resultados de la Unidad de Control. Ejemplos: relés, solenoides, contactores, display's, led, etc. (COLLAGUAZO).

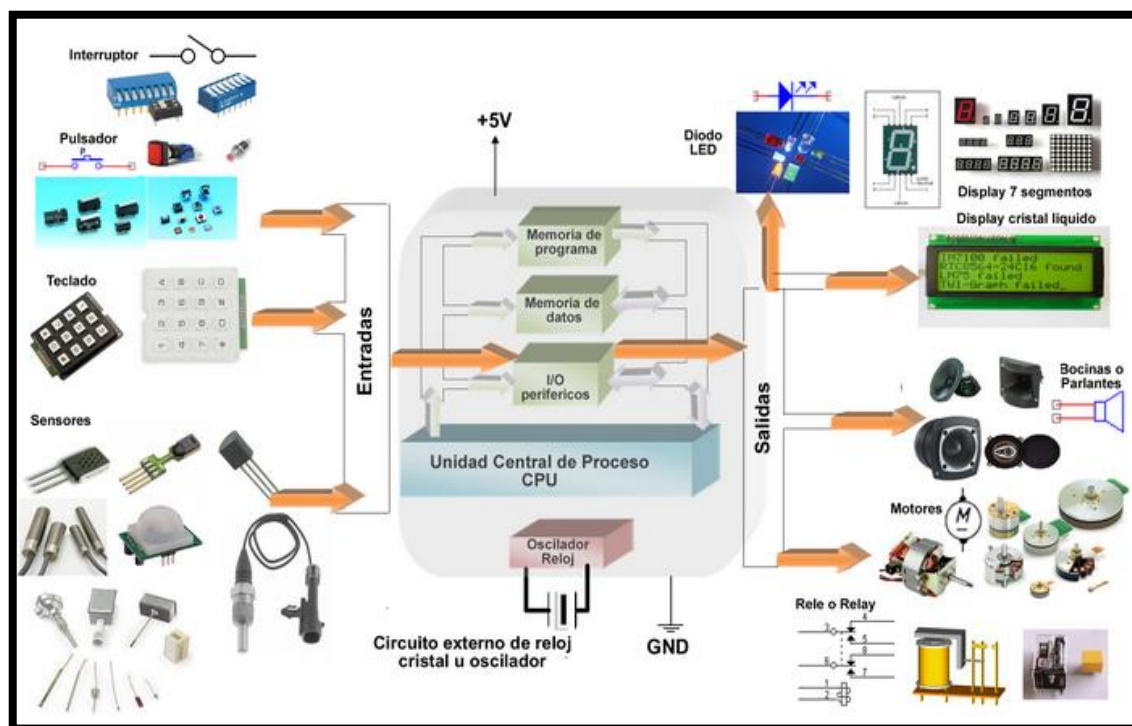


Figura 4 Sistema General

Fuente:

http://datateca.unad.edu.co/contenidos/309696/ECBTI_MPpyMC_V_02/leccin_17_sistemas_microcontrolados.html

2.1.4 Sistema Analógico

Un sistema analógico es aquel en el que sus señales son de tipo analógico. Sus componentes suelen trabajar en su zona lineal, en la que la relación que existe entre las señales de entrada y salida es constante, denominada zona de trabajo. Dichas señales pueden tomar cualquier valor dentro de unos límites determinados. (ACHA).

Ejemplo:

Un ejemplo de sistema analógico es el que se muestra en la Figura 5. Se trata de un medidor analógico de la velocidad de un motor.

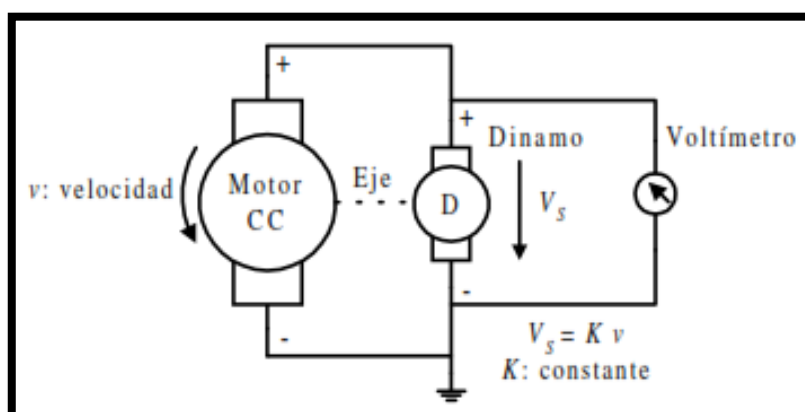


Figura 5 Ejemplo de sistema analógico

Fuente: ACHA, S. C., Manuel; PEREZ, Julio; RIOSEAS, Miguel. Electrónica Digital Introducción a la Lógica Digital Teoría, Problemas y Simulación.: Alfaomega Ra-Ma

2.1.5 Sistema Digital

Un sistema digital es aquel en el que sus señales son de tipo digital. Sus componentes trabajan en las zonas de saturación (sus señales de salidas no tienen una relación lineal respecto de sus entradas). Las señales de estos sistemas suelen ser próximas

a los potenciales de la alimentación, presentando dos estados diferenciados, correspondiendo cada uno de ellos a un nivel o valor de la magnitud binaria. (ACHA).

Ejemplo:

Un sistema digital es el que se muestra en la Figura 6, se trata de un cronómetro digital:

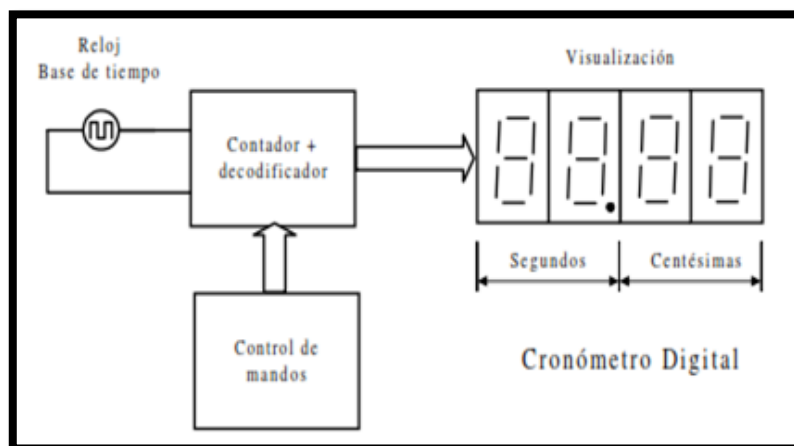


Figura 6 Ejemplo de sistema digital

Fuente: ACHA, S. C., Manuel; PÉREZ, Julio; RIOSERAS, Miguel. Electrónica Digital Introducción a la Lógica Digital Teoría, Problemas y Simulación.: Alfaomega Ra-Ma

2.1.6 Sistema Analógico-Digital

Un sistema analógico-digital es aquel en el que intervienen tanto señales analógicas como señales digitales; es decir, está compuesto de subsistemas analógicos y subsistemas digitales. (ACHA).

Ejemplo:

Un ejemplo de sistema analógico-digital es el que se muestra en la Figura 7, que corresponde a un reproductor de compact disc (CD). Un convertidor digital-analógico (D/A) transforma las señales digitales en analógicas, necesarias para excitar el altavoz, una vez hayan sido amplificadas en el bloque amplificador.

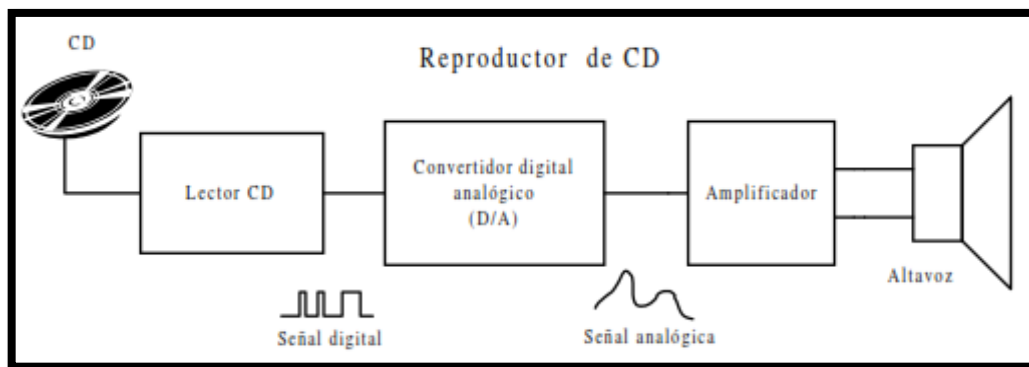


Figura 7 Ejemplo de sistema analógico-digital

Fuente: ACHA, S. C., Manuel; PEREZ, Julio; RIOSERAS, Miguel. *Electrónica Digital Introducción a la Lógica Digital Teoría, Problemas y Simulación.*: Alfaomega Ra-Ma

Diseño:

Un mismo diseño de sistema electrónico puede tener solución utilizando tanto técnicas digitales como analógica.

2.1.7 Ventajas de los sistemas digitales respecto a los analógicos

Principalmente en el procesamiento y transmisión de la información. Estas ventajas son:

- El número de operaciones básicas con variables digitales es muy reducido frente al número de operaciones con variables analógicas.
- Mayor precisión y versatilidad en el procesamiento de la información.
- Menor sensibilidad al ruido y alta fiabilidad.

- Alta capacidad de almacenamiento de información, pudiendo realizar accesos directos con nanosegundos).
- Posibilidad de detección y corrección de errores en la transmisión de información.
- Mayor flexibilidad en el sentido que pueden programarse con mayor facilidad para realizar cualquier algoritmo deseado.
- Capacidades de procesamiento superiores en términos de velocidad.
- La implantación de las funciones de almacenamiento y recuperación de información es más sencilla. (ACHA) (David).

2.2 SENSORES

Los sensores son dispositivos que miden la magnitud de una señal determinada y producen una señal relacionada. Los sensores usan las propiedades de los materiales de los que están hechos y así comparar su comportamiento ante las variaciones de la señal a medir, permiten obtener información del entorno. Por ejemplo, en caso de un sensor de temperatura que medirá temperatura mediante resistencia eléctrica, la cantidad que se mide es la temperatura y el sensor transforma una entrada de temperatura en un cambio en la resistencia. (STEREN).

Muchos de los sensores más usados son eléctricos o electrónicos, aunque existen de otros tipos, y las magnitudes a medir son fenómenos físicos interesantes como diversos tipos de energía, temperatura, intensidad lumínica, sonido, deformación, proximidad, presión, orientación, inclinación, humedad, aceleración, velocidad angular, tamaño, cantidad, etc. Como ejemplos podemos mencionar sensores de temperatura, humedad, fuerza, deformación, acidez, luz, sonido, contacto y proximidad. (STEREN).

Los transductores son dispositivos o elementos que al someterlos a un cambio físico experimenta un cambio relacionado. Con frecuencia se utiliza el término transductor de entrada en vez de sensor. Como ejemplos podemos mencionar transductores electroacústicos, electromecánicos, electromagnéticos, electroquímicos, fotoeléctricos, piezoeléctricos, termoelectrónicos y de presión. (STEREN).

Algunos ejemplos de aplicaciones de transductores son: los micrófonos, las bocinas, los teclados de los equipos, ventiladores, etc.

2.2.1 Características de operación

El funcionamiento y evaluación de los transductores se basa en diferentes características de operación, las características más consideradas son:

- **Intervalo.** Define los límites o el rango en magnitud entre los cuales puede variar la señal de entrada.
- **Extensión.** Es el valor máximo de entrada menos el valor mínimo que puede detectar un transductor.
- **Resolución.** Es la mínima señal de cambio en la señal de entrada detectada por el transductor.
- **Sensibilidad.** Es la relación que existe entre la salida y la entrada del transductor (salida/entrada).
- **Error.** Es la variación existente entre el valor real de la señal y el valor registrado por el transductor.

$$\text{Error} = \text{valor medido} - \text{valor real.}$$

- **Exactitud.** Es el grado hasta el cual un valor producido por un sistema de medición podría estar equivocado. La exactitud es igual a la suma de todos los errores posibles más el error en la exactitud de la calibración del transductor. Es la capacidad de reproducir la misma señal de salida a la misma señal real de entrada suponiendo un error constante del transductor.
- **Error por Histéresis.** Los transductores pueden producir distintas salidas de la misma cantidad medida considerando si los cambios en la señal de entrada son por incrementos o por decrementos de valor.
- **Error por no Linealidad.** En el intervalo de operación del transductor la relación entre la entrada y la salida es lineal, su grafica de salida en función de la entrada produce una línea recta. Sin embargo, pocos son los transductores que cumplen con esta función, al suponer la existencia de esta linealidad se producen errores. Este error se define como la desviación máxima respecto a la línea recta.
- **Estabilidad.** Es su capacidad para producir la misma salida cuando se usa para medir una entrada constante en un periodo.
- **Acoplamiento.** Se refiere a la impedancia de salida del transductor que afecta el circuito en el que se conecta. (BOLTON, 2006).

2.2.2 Factores a Considerar

En la tabla 1 se muestran los factores que hay que considerar para la elección de un sensor.

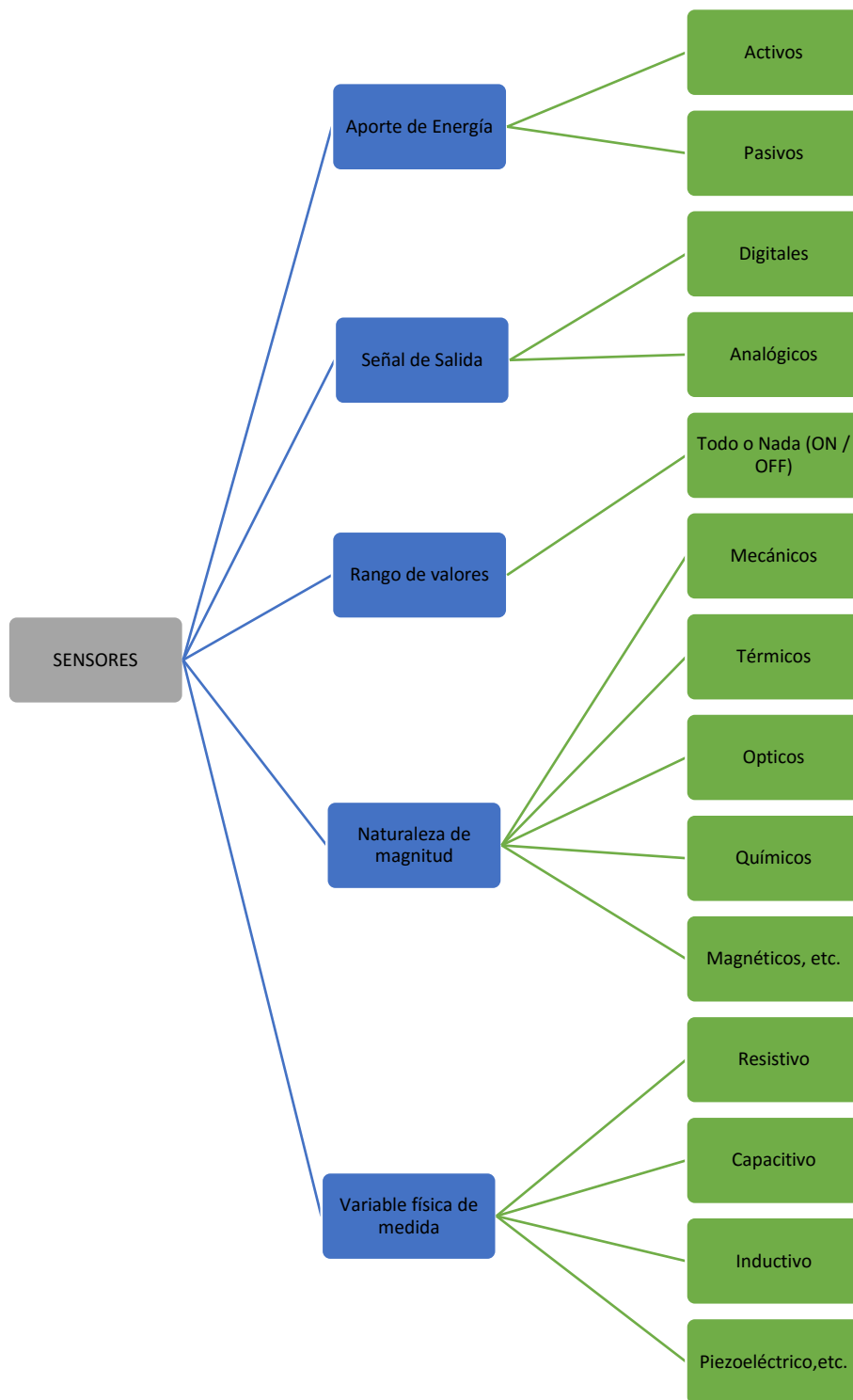
Tabla 1 *Factores a considerar en la elección de un sensor*

<i>Características de alimentación</i>	<i>Características ambientales</i>
Tensión	Margen de temperaturas
Corriente	Humedad
Potencia disponible	Vibraciones
Frecuencia (si alterna)	Agentes químicos
Estabilidad	¿Atmósfera explosiva? Entorno electromagnético
<i>Otros Factores</i>	
Peso	Longitud de cable necesaria
Dimensiones	Tiempo de conector
Vida media	Situación en caso de fallo
Costo de adquisición	Costo de verificación
Disponibilidad	Costo de mantenimiento
Tiempo de instalación	Costo de sustitución

Fuente: PALLAS, R. Sensores y Acondicionadores de señal (Cuarta ed.): Marcombo.

2.2.3 Clasificación de sensores

Los sensores se pueden clasificar según algunos parámetros:



Fuente: Autor

2.2.4 Tipos de sensores

Los sensores pueden ser de varios tipos, los más utilizados son:

- Según el aporte de energía, los sensores se pueden dividir en moduladores (activos) y generadores (pasivos).

Sensores activos.- la energía de la señal de salida procede de una fuente de energía externa o una señal de excitación para poder funcionar. Ejemplos: termistor, inductor, etc.

Sensores pasivos.- la energía de salida es suministrada por la entrada, sin necesidad de una fuente de energía externa. Ejemplos: termocuplas, sensores piezoeléctricos, etc.

- Según la señal de salida, los sensores se clasifican en analógicos y digitales.

Sensores Analógicos.- la señal de salida varía de forma continua tanto en amplitud, en el dominio temporal o de frecuencia. Generalmente proporcionan un valor de voltaje o corriente. Ejemplos: temperatura, intensidad lumínica, desplazamiento, etc.

Sensores Digitales.- la señal de salida varía en forma de escalones o estados discretos. Estos sensores proporcionan una señal codificada en pulsos o sistemas como BCD, binario, etc. No requieren de conversores A/D y la transmisión de su salida es más fácil. Ejemplos: Contacto (Switch), encoder, etc.

- Según el rango de valores, los sensores se clasifican en todo o nada (ON/OFF).

Sensores On/Off.- son los sensores que solo poseen dos estados, y que, estos estados, únicamente están separados por un umbral de la variable monitoreada. Ejemplos: para detectar presencia, posición, color, etc. (STEREN).

2.2.5 Sensores primarios

Se detallan los tipos de sensores más utilizados.

2.2.5.1 Sensores Resistivos

Este tipo de sensores están basados en la variación de la resistencia eléctrica, el potenciómetro es un elemento resistivo variable cuyo valor se determina por el desplazamiento de un contacto móvil deslizante o giratorio; el desplazamiento se convierte en una diferencia de potencial. (STEREN) .

Una aplicación de sensores resistivos son los detectores de temperatura que están basados en la variación de resistencia eléctrica de sus materiales y se denominan como RTD por sus siglas en inglés (Resistance Temperature Detectors). (STEREN).

2.2.5.2 Sensores Capacitivos

En este tipo de sensores sus materiales se comportan como condensadores eléctricos, los cuales sometidos a una diferencia de potencial adquieren una determinada carga eléctrica. Los condensadores están formados por dos placas o laminas conductoras separadas por un material dieléctrico. (STEREN).

Una aplicación de sensores capacitivos son los detectores de nivel, los cuales al modificar la cantidad de dieléctrico entre sus placas modifican sus características, dependiendo del nivel de líquido que estén midiendo. (STEREN).

2.2.5.3 Sensores Inductivos

Este tipo de sensores inductivos trabajan en forma de bobinas, cuyo funcionamiento es de almacenar energía en forma de campo magnético en función de la corriente eléctrica que pasa por sus terminales. (STEREN).

Una aplicación de este tipo de sensores es la detección de materiales ferrosos, por lo que son usados como detectores de posicionamiento, proximidad y como detectores de metales. (STEREN).

2.2.5.4 Sensores de luz

Los sensores de luz modifican sus propiedades de los materiales al variar la intensidad de luz que reciben.

Las aplicaciones de los sensores de luz son barreras fotoeléctricas (conteo y detección de personas u objetos), lectores de códigos de barras, controles automáticos de iluminación, etc. Las fotorresistencias y los fotodiodos son muy utilizados en este tipo de aplicaciones. (STEREN).

2.2.5.5 Sensores de flujo y caudal

Los sensores de flujo son dispositivos que pueden ser instalados en tuberías, canales, conductos abiertos o cerrados; permiten determinar si hay circulación de un gas o fluido. El caudal es la cantidad de material, en peso o volumen, que fluye por unidad de tiempo. (PALLAS).

2.2.5.6 Sensores de Efecto Hall

El efecto Hall consiste en la producción de una caída de voltaje a través de un conductor o semiconductor con corriente, bajo la influencia de un campo magnético externo. Para esto es necesario que la dirección del campo magnético sea perpendicular a la dirección de flujo de la corriente. (Sensores de efecto Hall). Como consecuencia del campo magnético, los electrones que se desplazan se desvían hacia un lado de la placa que se carga negativamente, mientras el lado opuesto se carga positivamente ya que los electrones se alejan. El resultado es una diferencia de potencial conocida como Voltaje Hall. (BOLTON, 2006).

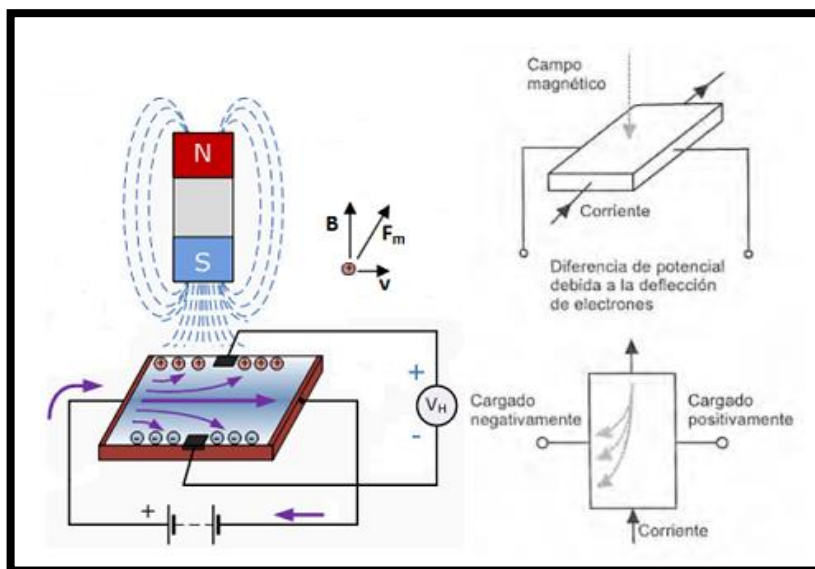


Figura 8 Efecto Hall

Fuente: BOLTON, W. (2006). Sistemas de Control Electrónico en la Ingeniería Mecánica y Eléctrica (Tercera ed.): Alfaomega

Generalmente los sensores de efecto Hall están incorporados en circuitos integrados para procesar señales.

Básicamente existen dos tipos de este sensor:

1. Tipo lineal.- donde la salida varía de manera razonablemente lineal con la densidad de flujo magnético.
2. Tipo umbral (salida digital).- donde la salida cae en forma brusca cuando se presenta cierta densidad de flujo magnético. (BOLTON, 2006).

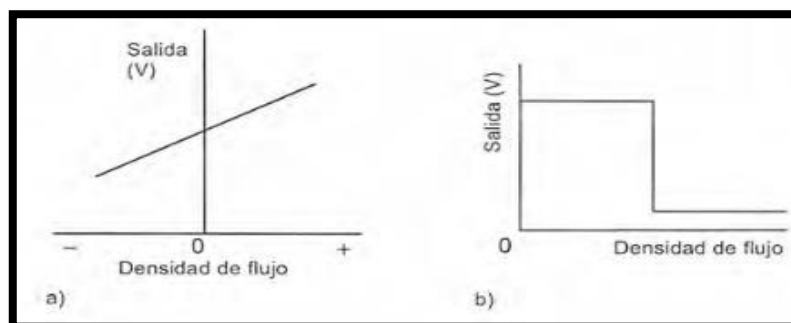


Figura 9 Sensores de efecto Hall: a) lineal; b) de umbral

Fuente: BOLTON, W. (2006). Sistemas de Control Electrónico en la Ingeniería Mecánica y Eléctrica (Tercera ed.): Alfaomega

Entre las aplicaciones más comunes de este tipo de sensores son: medición de corriente, encendido electrónico (autos, motos), sistemas de frenos ABS, bloqueo de cerraduras, medición de velocidad, nivel de líquidos, sensado de posición, etc.

2.2.5.7 Interruptores

Los interruptores son dispositivos que pueden ser considerados como sensores, aunque no modifican las propiedades físicas de sus componentes, pueden detectar diferentes estados de fenómenos u objetos.

Los interruptores pueden ser utilizados para aplicaciones donde se deba detectar movimientos, posiciones y frecuencia de comportamiento de estos objetos. (STEREN).

2.3 ACTUADORES

Son elementos de los sistemas de control que transforman la salida de un microprocesador o un controlador en una acción de control para una maquina o dispositivo. (BOLTON, 2006). Los actuadores actúan en forma de señal neumática, electrónica, digital, óptica, hidráulica o por radio. (CREUS, 2011). Existen algunos tipos de actuadores.

2.3.1 Actuadores Neumáticos

Las señales neumáticas se utilizan para manejar elementos finales de control, incluso cuando el sistema de control es eléctrico. Se debe a que con esas señales es posible

accionar válvulas grandes y otros dispositivos de control que requieran mucha potencia para mover cargas considerables. (BOLTON, 2006).

Los actuadores neumáticos utilizan aire comprimido por un compresor y la transforman en energía mecánica por medio de pistones o turbinas. (KUMAR, 2010).

2.3.1.1 Ventajas

- Son económicos, los componentes son perfectamente asequibles.
- El aire comprimido es limpio, lo que no tiene riesgo de explosiones e insensibles a fluctuaciones de temperatura.
- Reducido costo de mantenimiento.
- Tienen una acción y tiempo de respuesta muy rápida. (KUMAR, 2010).

2.3.1.2 Desventajas

- Dificultad en el control de la precisión en la velocidad y la posición.
- No son convenientes utilizados para mover cargas pesadas bajo control preciso.
- Su sistema es ruidoso. (KUMAR, 2010).

2.3.2 Actuadores Hidráulicos

Las señales hidráulicas se usan en dispositivos de control de mucha mayor potencia, son más costosas que los sistemas neumáticos. (BOLTON, 2006).

Los actuadores neumáticos utilizan su fuente de energía de aceites minerales a una presión comprendida entre los 50 y 100 bares, y la transforman en energía mecánica por medio de pistones o turbinas. (KUMAR, 2010).

La precisión obtenida es mayor que la de los actuadores neumáticos.

2.3.2.1 Ventajas

- Alta relación y eficiencia entre potencia-tamaño.
- Es posible el control completo y preciso en la velocidad y posición.
- Mayor capacidad de carga que los actuadores eléctricos y neumáticos.
- Son auto lubricantes y no corrosivos. (KUMAR, 2010).

2.3.2.2 Desventajas

- Pueden presentarse fugas que pueden causar riesgo de incendios, pérdida de rendimiento y contaminación del área.
- La fuente de alimentación puede ser ruidosa.
- Complejo mantenimiento. (KUMAR, 2010).

2.3.3 Actuadores eléctricos

Estos actuadores presentan gran control, sencillez y precisión, por lo tanto son los más utilizados.

Los actuadores entre los más comunes tenemos: motores de corriente continua (DC), motores de corriente (AC), motores paso a paso, servo motores, alambres musculares, led's, buzzer, solenoide, etc.

2.3.3.1 Ventajas:

- Amplia disponibilidad en el suministro de energía.
- No contaminan el área de trabajo.
- Fácil mantenimiento y reparación.
- Pueden ser de peso ligero. (KUMAR, 2010).

2.3.3.2 Desventajas:

- Debido a la complejidad más alta del sistema de transmisión, se generan costos adicionales para su adquisición.
- No son intrínsecamente seguros. (KUMAR, 2010).

2.3.4 Características importantes a considerar

Hay que considerar algunas características importantes antes de seleccionar un actuador, como por ejemplo:

- Potencia.
- Facilidad de control.
- Repetitividad.
- Material / resistencia.

- Peso y volumen.
- Precisión.
- Velocidad.
- Mantenimiento.
- Coste.(MINA, 2015).

2.4 ÁREAS DE APLICACIÓN DE LOS SENSORES Y ACTUADORES

- Tanto sensores como actuadores son utilizados en diferentes áreas de la ciencia:
- Industria automotriz
- Industria aeroespacial
- Medicina
- Industria de manufactura
- Robótica
- Ingeniería Eléctrica
- Ingeniería Mecánica
- Ingeniería Aeroespacial.
- Bioelectrónica
- Ingeniería de materiales
- Geología
- Ingeniería Espacial
- Ingeniería Mecatrónica
- Física, etc. (CREUS, 2011).

2.5 ESTUDIO DE LOS DIFERENTES DISPOSITIVOS DE PROPÓSITO GENERAL

El desarrollo del presente proyecto está basado en dos plataformas de hardware y software libre: el microcontrolador Arduino y la minicomputadora.

Ambos dispositivos deben disponer de unas especificaciones básicas: pequeño tamaño, bajo coste, silenciosos y con unos consumos de energía reducidos. (ENRIQUEZ, 2009).

2.5.1 Arduino.



Figura 10 Logo de Arduino
Fuente: <https://www.arduino.cc>

Arduino es una plataforma de prototipos electrónicos de código abierto (open source) basada en hardware y software flexibles y fáciles de usar. El microcontrolador de la placa de Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). (Robotica, 2012).

Arduino se puede utilizar para crear objetos interactivos, leyendo datos de una gran variedad de sensores (interruptores, sensores, etc.) y controlar gran variedad de tipos de actuadores físicos (luces, motores, etc.).

Los proyectos de Arduino pueden ser autónomos o pueden comunicarse con un programa (software) que se ejecute en el ordenador. (Robotica, 2012).

2.5.1.1 ¿Por qué Arduino?

Arduino simplifica el proceso de trabajo con microcontroladores, ofreciendo algunas ventajas respecto a otros sistemas a profesores, estudiantes y amateurs:

- **Asequible.** Las placas de Arduino son relativamente baratas comparadas con otras plataformas de microcontroladores.
- **Multi-plataforma.** El software de Arduino funciona en los sistemas operativos Windows, Macintosh OS X y Linux.
- **Entorno de programación simple y claro.** El entorno de programación de Arduino es fácil de usar para principiantes y lo suficientemente flexible para los usuarios avanzados.
- **Código abierto y software extensible.** El software Arduino está publicado bajo licencia de código abierto y preparado para ser ampliado por programadores experimentados. El lenguaje puede ampliarse mediante librerías de C++, y si está interesado en profundizar o entender los detalles técnicos pueden hacer un salto a la programación en el lenguaje AVR C en el cual está basado.
- **Código abierto y hardware extensible.** Arduino está basado en los microcontroladores ATMEGA de Atmel. Los planos para los módulos están publicados bajo licencia Creative Commons, por lo que diseñadores de circuitos con experiencia pueden hacer su propia versión del módulo, extendiéndole y mejorándole. (ENRIQUEZ, 2009).

2.5.2 Alternativas del microcontrolador Arduino

Existen diferentes modelos de Arduino para ser utilizados dependiendo de la aplicación que se desee realizar, con diferentes formatos y configuraciones de hardware.

2.5.2.1 Arduino Uno

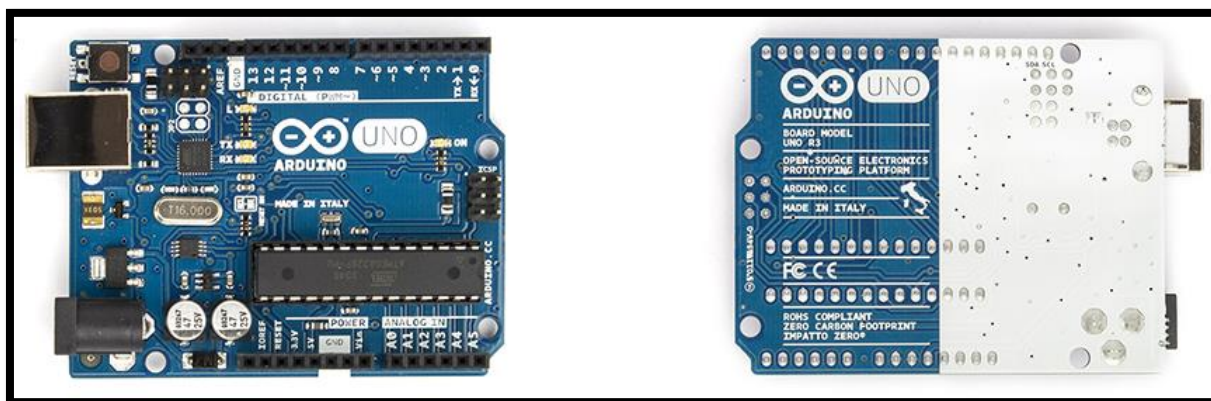


Figura 11 Parte frontal y trasera de Arduino Uno
Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardUno>

El Arduino Uno es una placa electrónica basado en el microcontrolador Atmega328. Esta placa posee 14 pines digitales de entrada/salida (6 pines se pueden utilizar como salidas PWM), 6 pines de entradas analógicas, un resonador cerámico a 16 MHz, un conector USB, un Jack de poder y un botón de reinicio. Además el Arduino Uno viene con periféricos internos capaces de soportar protocolos de comunicación como el UART, I2C y SPI. (Arduino).

Para que el microcontrolador funcione basta con conectarlo a un ordenador con cable USB o a través de un adaptador de CA o una batería de CC.

Tabla 2 Resumen de las características técnicas de Arduino

Características	Descripción
Microcontroladores	Atmega328
Tensión de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (limites)	6-20V
Digital pines I/O	14 (de las cuales 6 proporcionan salida PWM)
Botones de entrada analógica	6
Corriente DC por E/S Pin	40 mA
Corriente DC de 3.3V Pin	50 mA
Memoria Flash	32 KB (ATmega328) de los cuales 0.5 KB utilizado por el gestor de arranque
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Velocidad de reloj	16 MHz
Longitud	68.6 mm
Anchura	53.4 mm
Peso	25 g

Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardUno>

2.5.2.2 Arduino Leonardo

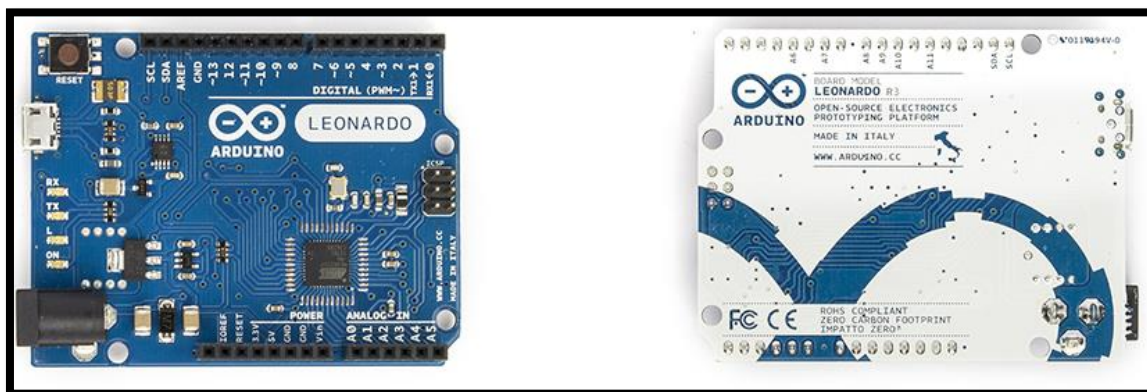


Figura 12 Parte frontal y trasera de Arduino Leonardo
Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardLeonardo>

El Arduino Leonardo es una placa electrónica basado en el microcontrolador Atmega32u4. Esta placa posee 20 pines digitales de entrada/salida (7 pines se pueden utilizar como salidas PWM y 12 pines como entradas analógicas), un oscilador de cristal de 16 MHz, un conector micro USB, un conector de alimentación y un botón de reinicio. Además el Arduino Leonardo también es compatible con protocolos de comunicación como el UART, I2C y SPI. (Arduino).

Para que el microcontrolador funcione basta con conectarlo a un ordenador con cable USB o a través de un adaptador de CA o una batería de CC.

Tabla 3 *Resumen de las características técnicas de Arduino Leonardo*

Características	Descripción
Microcontroladores	Atmega32u4
Tensión de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (limites)	6-20V
Digital pines I/O	20
Canales PWM	7
Canales de entrada analógica	12
Corriente DC por E/S Pin	40 mA
Corriente DC de 3.3V Pin	50 mA
Memoria Flash	32 KB (ATmega32u4) de los cuales 4 KB utilizado por el gestor de arranque
SRAM	2.5 KB (ATmega32u4)
EEPROM	1 KB (ATmega32u4)
Velocidad de reloj	16 MHz
Longitud	68.6 mm
Anchura	53.3 mm
Peso	20 g

Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardLeonardo>

2.5.2.3 Arduino Mega 2560

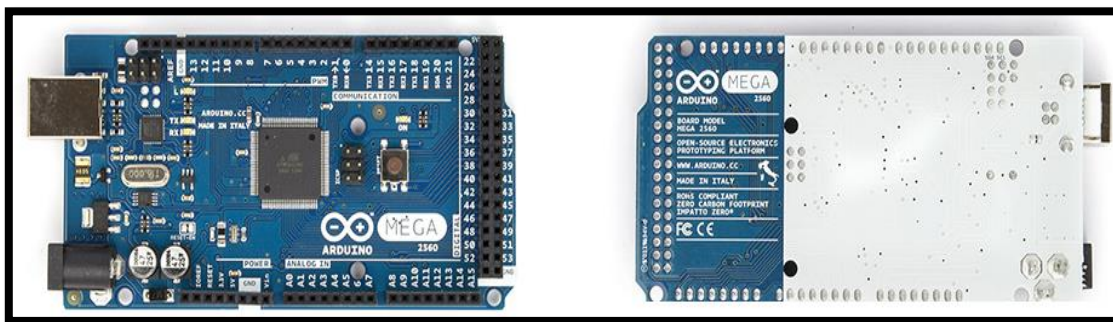


Figura 13 Parte frontal y trasera de Arduino Mega 2560
Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardMega2560>

El Arduino Mega 2560 es una placa electrónica basado en el microcontrolador Atmega2560. Esta placa cuenta con 54 pines digitales de entrada/salida (15 pines se pueden utilizar como salidas PWM), 16 pines como entradas analógicas, 4 puertos seriales (UART), un oscilador de cristal de 16 MHz, un conector USB, un conector de alimentación y un botón de reinicio. Además el Arduino Mega 2560 también es compatible con protocolos de comunicación como el UART, I2C y SPI. (Arduino).

Para que el microcontrolador funcione basta con conectarlo a un ordenador con cable USB o a través de un adaptador de CA o una batería de CC.

Tabla 4 Resumen de las características Técnicas de Arduino Mega

Características	Descripción
Microcontroladores	Atmega2560
Tensión de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (limites)	6-20V
Digital pines I/O	54 (de las cuales 15 proporcionan salida PWM)

Canales de entrada analógica	16
Corriente DC por E/S Pin	40 mA
Corriente DC de 3.3V Pin	50 mA
Memoria Flash	256 KB de los cuales 8 KB utilizado por el gestor de arranque
SRAM	8 KB
EEPROM	4 KB
Velocidad de reloj	16 MHz

Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardMega2560>

2.5.2.4 Arduino DUE

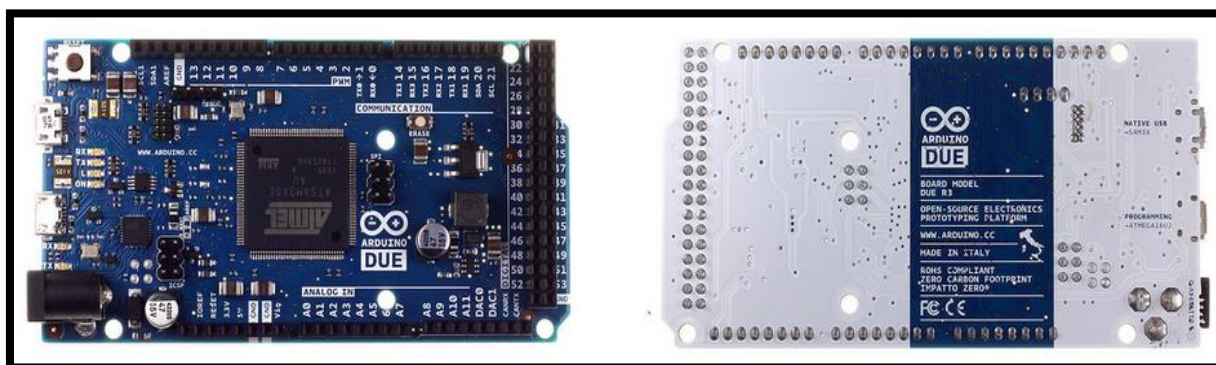


Figura 14 Parte frontal y trasera de Arduino Due.

Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardDue>

El Arduino Due es una placa electrónica basado en el microcontrolador Atmel SAM3X8E ARM Cortex-M3 de la CPU. Es la primera placa Arduino basado en un microcontrolador núcleo ARM de 32 bits. Esta placa cuenta con 54 pines digitales de entrada/salida (12 pines se pueden utilizar como salidas PWM), 12 pines como entradas analógicas, 4 puertos seriales (UART), un reloj de 84 MHz, un conector USB OTG, 2 DAC (convertidores Digital/Analógico), 2 protocolos de comunicación I2C, un conector de alimentación, una cabecera SPI, un botón de reinicio y un botón de borrado. A diferencia de

otras placas de Arduino esta placa puede tolerar la tensión máxima de 3.3V en los pines de entrada/salida. Si se alimenta con voltajes más altos podría dañar la placa. (Arduino).

Además el Arduino Due también es compatible con protocolos de comunicación como el UART, I2C y SPI. Para que el microcontrolador funcione basta con conectarlo a un ordenador con cable USB o a través de un adaptador de CA o una batería de CC.

Tabla 5 *Resumen de las características Técnicas de Arduino Due*

Características	Descripción
Microcontroladores	AT91SAM3X8E
Tensión de funcionamiento	3.3V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (limites)	6-16V
Digital pines I/O	54 (de las cuales 12 proporcionan salida PWM)
Botones de entrada analógica	12
Salida analógica	2 (DAC)
Total Corriente de salida DC en todas las líneas E/S	130 mA
Corriente DC de 3.3V Pin	800 mA
Corriente DC de 5V Pin	800 mA
Memoria Flash	512 KB disponibles para las aplicaciones de usuario
SRAM	96 KB
Velocidad de reloj	84 MHz
Longitud	101.52 mm
Anchura	53.3 mm
Peso	36 g

Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardDue>

2.5.3 Tabla comparativa de los Arduinos mencionados

En la tabla 6 se observan las características de las placas de Arduino que fueron tomadas en cuenta para el proceso de selección del microcontrolador. (Arduino).

Tabla 6 *Comparativa entre Arduinos*

Característica de Arduino	UNO	Mega 2560	Leonardo	DUE
Tipo de microcontrolador	Atmega 328	Atmega 2560	Atmega 32U4	AT91SAM3X8E
Velocidad de reloj	16 MHz	16 MHz	16 MHz	84 MHz
Pines digitales de E/S	14	54	20	54
Entradas analógicas	6	16	12	12
Salidas analógicas	0	0	0	2 (DAC)
Memoria de programas (Flash)	32Kb	256Kb	32Kb	512Kb
Memoria de datos (SRAM)	2Kb	8Kb	2.5Kb	96KB
Memoria auxiliar (EEPROM)	1Kb	4Kb	1KB	0Kb
Voltaje Operación/Voltaje Entrada	5V/7-12V	5V/7-12V	5V/7-12V	3.3V/7-12V
EEPROM (KB)	1	4	1	0
USB	Regular	Micro	Micro	2 Micro
UART	1	4	1	4

Fuente: <http://www.arduino.cc/en/Products.Compare>

2.5.4 MINICOMPUTADORA

Las mini PC o minicomputadoras son placas de hardware y software libre (open source) capaces de alojar un sistema operativo, en general una distribución Linux adaptada, y disponen de los principales puertos como cualquier otra computadora Ethernet, USB o mini USB, etc. Algunas placas tienen una capacidad de tener integrada una tarjeta WIFI,

otras placas poseen salida HDMI, cada una de las minicomputadoras poseen sus propias características. Además existen algunas placas que destacan por su capacidad de memoria, otras por su capacidad gráfica. (Ultra-lab).

Los objetivos de estos minicomputadores son fomentar la enseñanza de ciencias de la computación y que países pobres puedan contar con un PC para cada niño. (Isaac).

2.5.4.1 INTEL GALILEO

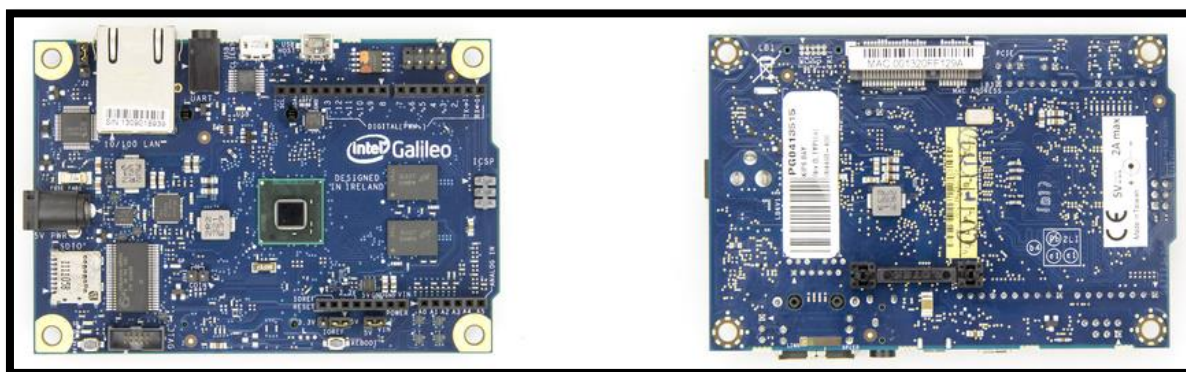


Figura 15 Parte frontal y trasera de Intel Galileo
Fuente: <http://www.arduino.cc/en/ArduinoCertified/IntelGalileo>

La placa Intel Galileo Gen 2 es la primera de una familia de placas de desarrollo y prototipos certificados por Arduino, basadas en la arquitectura Intel. Ofrece a los usuarios un entorno de desarrollo de hardware y software de código abierto, complementa y amplía los productos de Arduino para ofrecer funciones de computación más avanzadas. (INTEL).

Los pines 0 a 13 son digitales, los pines 0 a 5 son entradas analógicas, posee un conector de fuente de poder, los pines 0 y 1 son del puerto serial (UART), todos los pines

mencionados se encuentran situados de la misma manera que en el Arduino Uno R3. (Arduino).

Características:

Galileo está basado en el procesador de aplicaciones Intel® Quark™ SoC X1000, una arquitectura de conjunto de instrucciones de procesador Intel® Pentium® de 32 bits. La placa Galileo es compatible con una amplia variedad de interfaces de E/S estándar en la industria, entre ellas la ranura mini-PCI Express de tamaño completo, el puerto Ethernet de 10/100 Mb, la ranura microSD, el host USB y el puerto cliente USB.

También puede ser programada a través del entorno de desarrollo integrado (IDE) de software Arduino, que es compatible con los sistemas operativos host Microsoft Windows*, Mac OS* y Linux. (INTEL).

Galileo también soporta protocolos de comunicación como UART, I2C y SPI. Esta placa es la primera que proporciona una ranura mini PCI Express (mPCIe), donde cualquier módulo mPCIe estándar se pueda conectar y utilizar para proporcionar aplicaciones tales como WIFI, Bluetooth o conectividad celular. (Arduino).

2.5.4.2 RASPBERRY PI +B

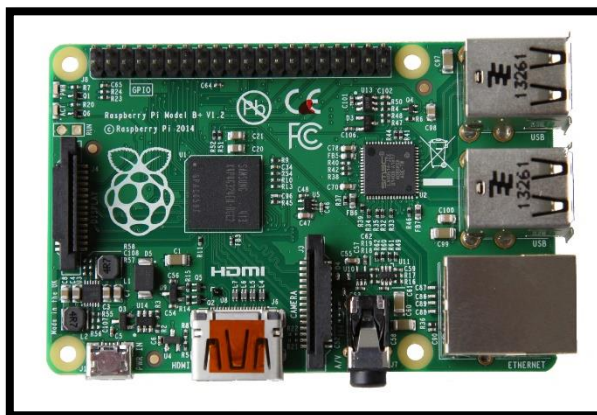


Figura 16 Parte frontal de Raspberry PI
Fuente: www.raspberrypi.org

La tarjeta Raspberry Pi es un minicomputador del tamaño de una tarjeta de crédito de bajo costo, el objetivo de la fundación Raspberry Pi es promover la educación de adultos y niños de todo el mundo para aprender a programar y entender cómo funcionan los ordenadores. (RaspberryPI).

Raspberry está basado en el sistema operativo Linux en su versión Debian con la distribución Raspbian Wheezy recomendada por la fundación Raspberry Pi. (RaspberryPI).

Características:

A continuación se describe las especificaciones técnicas más importantes del fabricante:

- Procesador (con gráfica integrada): Broadcom BCM2835. Contiene ARM1176JZFS, con unidad de coma flotante, funciona a 700Mhz y un Videocore 4 GPU.

- Memoria Ram: 512 MB
- Características Técnicas de la GPU:
 - La GPU es capaz de mover contenidos con calidad Blu-ray, usando H.264 hasta 40MBits/s. Dispone un Core 3D con soporte para las librerías OpenGL ES2.0 y OpenVG. Es capaz de decodificar 1080p30 H.264 high-profile.
- Dispositivo de Arranque:
 - Memoria SD card.
- Conectores:
 - 2x Conectores USB 2.0, Conector Ethernet RJ-45 10/100
 - Salida de Video Digital HDMI (Audio y Video)
 - Salida de Video Analógico (S-Video)
 - Audio Analógico (Conector 3,5mm)
 - Conector GPIO
 - Conector de alimentación Micro USB
 - Lector de memorias SD (Utilizado para arrancar el dispositivo)
- Alimentación:
 - Vía Micro USB 5 Voltios
- Sistemas Operativos Soportados:
 - Raspbian “wheezy” (Debian), ArchLinux, Fedora, QtonPi (OTSDK), Open Elec, Raspbcm, Android (en desarrollo por usuarios). (SILVA, 2013).

2.5.4.3 PCDUINO V1.

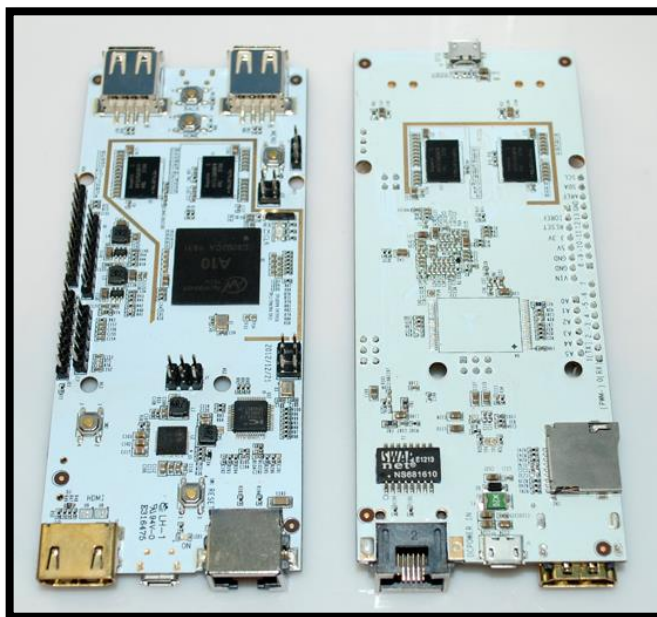


Figura 17 Parte frontal y trasera de Pcduino V1
Fuente: http://www.linksprite.com/?page_id=803

Pcduino es una plataforma de mini PC que combina los beneficios de un mini PC ARM basado en Arduino, Pcduino se define simplemente como un mini PC + Arduino. (LinkSprite).

La importancia de unir estos dos mundos open-hardware, los mini PCs y los Arduinos, radica en el aprovechamiento de la gran cantidad de desarrollo que existe, tanto de hardware como de software Arduino, para incorporarlos y gestionarlos desde el mini PC, con todo lo que este aporta para desarrollo, visualización, comunicaciones, almacenamiento, etc. Así se benefician ambos ecosistemas informáticos, y se puede decir que la unión es más que la suma de las partes. (ASANZA).

Este enlace se puede conseguir por medio de tarjetas adaptadoras que hacen de puente entre el mini PC base y todos los dispositivos y sensores usados con Arduino. Unos 'drivers' o controladores apropiados instalados en el pc se encargarán de emular la operatividad Arduino del microcontrolador Atmel, pasando esa tarea al propio procesador del mini PC. (ASANZA).

Características:

La placa cuenta con las siguientes características principales:

- CPU: 1GHz ARM Cortex A8
- GPU: OpenGL ES2.0, OpenVG 1.1 Mali 400 Core
- DRAM: 1GB
- Almacenamiento: 2GB Flash y slot de memoria SD expandible hasta 32GB
- Salida de Video: HDMI
- Sistemas Operativos compatibles: Linux3.0 + Ubuntu 12.04, Android ICS 4.0
- Requerimiento de alimentación: 5VDC, 2A
- 2 puertos USB
- 1 puerto Ethernet 10/100Mbps
- Lenguajes de programación: C++, Python (Ubuntu) y Java (Android).
- I/O:
 - ✓ 14 pines digitales GPIO
 - ✓ 1 UART RX
 - ✓ 1 UART TX
 - ✓ 2 pines PWM
 - ✓ 6 pines ADC

- ✓ 4 pines SPI
- ✓ 2 pines I2C. (LinkSprite).

2.5.5 Tabla comparativa de las minicomputadoras mencionadas

En la tabla 7 se puede observar una tabla comparativa con las características más importantes de las minicomputadoras.

Tabla 7 Comparativa de las minicomputadoras

Características	INTEL GALILEO GEN 2	PCDUINO LITE	RASPBERRY PI B+
Procesador	400 Mhz x86 Quark	1 Ghz ARM Cortex A8 CPU	BCM2835 ARM11 700 Mhz
Gráficos	N/A	OpenGL ES2.0, OpenVG 1.1 Mali 400 core	Broadcom VideoCore IV
RAM	256 MB DDR3	1 GB DRAM	512 MB
Almacenamiento	Tarjeta SD	2 GB Flash	Tarjeta SD
Tamaño	4.88" X 2.84"	4.93 " x 2.05"	3.37" x 2.125"
Voltaje de Entrada	5V	5V	5V
Interfaces	GPIO, UART, I2C, SPI	Digital GPIO 14	GPIO, UART, I2C, SPI
USB 2.0	MicroUSB, USB	MicroUSB, USB	MicroUSB, USB
Dispositivo USB	USB Host	Cliente Host	USB Host
Ranuras de almacenamiento	MicroSD	MicroSD	MicroSD
Ethernet	SI	SI	SI
WI-FI	NO	SI	SI
Entrada de micrófono	NO	NO	NO
Salida de Audio	NO	NO	SI
Entradas Analógicas	SI (6)	SI (12)	NO
HDMI	NO	SI	SI

Fuente: Arduino. Intel Galileo Gen2. <https://www.arduino.cc/en/ArduinoCertified/IntelGalileoGen2>
LinkSprite, T. pcDuino vs. Raspberry Pi vs. Beagle Bone vs. Arduino. http://www.linksprite.com/?page_id=1063

2.6 PROTOCOLOS DE ACCESO A LA RED

2.6.1 Protocolos y comunicaciones de red

El IEEE es una de los organismos de estandarización líderes en el mundo. “Los estándares IEEE 802.3 e IEEE 802.11 son estándares IEEE importantes en redes de computadoras. El estándar IEEE 802.3 define el control de acceso al medio (MAC) para Ethernet por cable. Esta tecnología generalmente es para las LAN, pero también tiene aplicaciones para redes de área extensa (WAN). El estándar 802.11 define un conjunto de estándares para implementar redes de área local inalámbricas (WLAN). Este estándar define el MAC físico y de enlace de datos del modelo de interconexión de sistema abierto (OSI) para las comunicaciones inalámbricas.” (CISCO). Existen dos tipos básicos de modelos de red:

2.6.1.1 Modelo de protocolo TCP/IP

Modelo de protocolo: el modelo TCP/IP representa la funcionalidad requerida para interconectar la red humana con la red de datos. Describe las funciones que tienen lugar en cada capa de protocolo dentro de una suite TCP/IP. (CISCO).

2.6.1.2 Modelo de referencia OSI

Modelo de referencia: este tipo de modelo describe que es lo que se debe realizar en cada capa determinada, pero sin regir la forma en que se debe lograr. El modelo OSI es

el más conocido, y ayuda a lograr un mejor entendimiento de las funciones y procesos involucrados. (CISCO).

En la figura 18 se observa los dos modelos principales que se utilizan al hablar de funcionalidad de red TCP/IP y OSI.

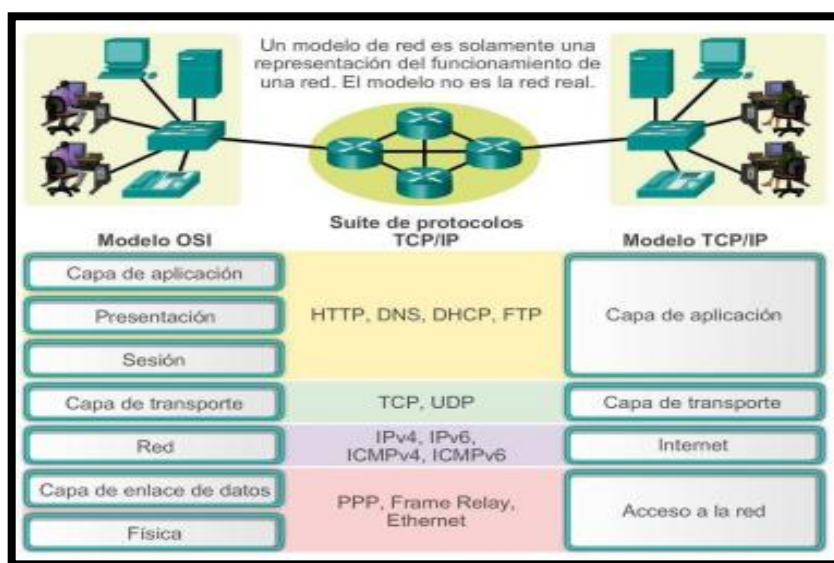


Figura 18 Modelos de Red

Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5

2.6.2 Medios para acceder a la red

Existen tres formatos básicos de medios de red.

- Cable de cobre: las señales son patrones de pulsos eléctricos.
- Cable de fibra óptica: las señales son patrones de luz.
- Conexión inalámbrica: las señales son patrones de transmisiones de microondas.

(CISCO).

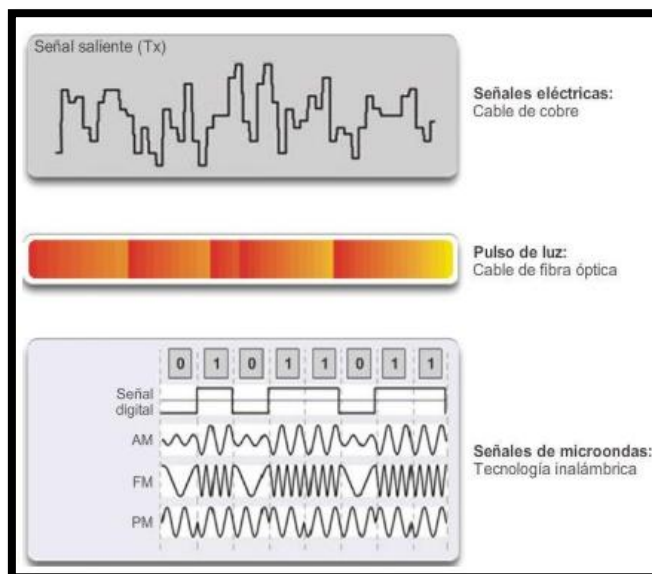


Figura 19 Ejemplos de señalización para medios inalámbricos, de cobre y de fibra óptica
Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5

2.6.3 Propiedades de los medios inalámbricos

“Los medios inalámbricos transportan señales electromagnéticas que representan los dígitos binarios de las comunicaciones de datos mediante frecuencias de radio y de microondas.” (CISCO)

Algunas propiedades importantes de los medios inalámbricos son:

- **Área de cobertura:** las tecnologías inalámbricas funcionan bien en entornos abiertos. Sin embargo, existen determinados materiales que limitan la cobertura efectiva.
- **Interferencia:** la tecnología inalámbrica también es vulnerable a la interferencia y también puede ser afectada por dispositivos comunes como teléfonos inalámbricos domésticos, algunos tipos de luces fluorescentes, hornos de microondas, etc.
- **Seguridad:** dispositivos y usuarios sin autorización pueden acceder a la red. En consecuencia, la seguridad de la red es fundamental en la administración de una red inalámbrica. (CISCO).

2.6.3.1 Tipos de medios inalámbricos

En la actualidad existen tres estándares comunes de comunicación de datos en medios inalámbricos:

2.6.3.1.1 Estándar IEEE 802.11

“Estándar IEEE 802.11: la tecnología de LAN inalámbrica (WLAN), comúnmente denominada “Wi-Fi”, utiliza un sistema por contienda o no determinista con un proceso de acceso múltiple por detección de portadora y prevención de colisiones (CSMA/CA) para acceder a los medios. (CISCO).

2.6.3.1.2 Estándar IEEE 802.15

Estándar IEEE 802.15: el estándar de red de área personal inalámbrica (WPAN), comúnmente denominado “Bluetooth”, utiliza un proceso de emparejamiento de dispositivos para comunicarse a través de distancias de 1 a 100 m. (CISCO).

2.6.3.1.3 Estándar IEEE 802.16

Estándar IEEE 802.16: conocido comúnmente como “interoperabilidad mundial para el acceso por microondas” (WiMAX), utiliza una topología de punto a multipunto para proporcionar acceso a servicios de banda ancha inalámbrica.” (CISCO).



Figura 20 Diferencias entre los medios inalámbricos
Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5

2.6.4 LAN inalámbrica

Las tecnologías de red inalámbricas 802.11 utilizan el método CSMA/CA (Acceso Múltiple por detección de portadora y prevención de colisiones), el cual el dispositivo analiza los medios para detectar la presencia de una señal de datos. Si el medio está disponible, el dispositivo envía una notificación a través del medio, sobre la intención de utilizar el medio. Luego de recibir la autorización el dispositivo envía los datos. (CISCO).

Una WLAN requiere de los siguientes dispositivos:

- Punto de acceso inalámbrico: el punto de acceso (AP) inalámbrico concentra las señales inalámbricas de los usuarios y se conecta a la infraestructura de red existente basada en medios de cobre, como Ethernet.

- Adaptadores de NIC inalámbricas: proporcionan capacidad de comunicación inalámbrica a cada host de red. (CISCO).

2.6.4.1 Estándares WI-FI 802.11

Los distintos estándares 802.11 evolucionaron con los años. Los estándares incluyen:

- IEEE 802.11a: opera en la banda de frecuencia de 5 GHz y proporciona velocidades de hasta 54 Mb/s. Posee un área de cobertura menor y es menos efectivo ya que opera en frecuencias superiores. Los dispositivos que funcionan conforme a este estándar no son interoperables con los estándares 802.11b y 802.11g.
- IEEE 802.11b: opera en la banda de frecuencia de 2,4 GHz y proporciona velocidades de hasta 11 Mb/s. Los dispositivos que implementan este estándar tienen un mayor alcance que el estándar 802.11a.
- IEEE 802.11g: opera en la banda de frecuencia de 2,4 GHz y proporciona velocidades de hasta 54 Mbps. Los dispositivos que implementan este estándar operan en la misma radiofrecuencia y tienen un alcance de hasta 802.11b pero con un ancho de banda de 802.11a.
- IEEE 802.11n: opera en la banda de frecuencia de 2,4 GHz y 5 GHz. Y proporcionan velocidades de 150 Mb/s a 600 Mb/s, con un alcance de hasta 70 m. Es compatible con dispositivos 802.11a, b y g.
- IEEE 802.11ac: opera en la banda de 5 GHz y proporciona velocidades de datos que van de 450 Mb/s a 1,3 Gb/s (1300 Mb/s); es compatible con dispositivos 802.11a/n.

- IEEE 802.11ad: también conocido como “WiGig”. Utiliza una solución de Wi-Fi de triple banda con 2,4 GHz, 5 GHz y 60 GHz, y ofrece velocidades teóricas de hasta 7 Gb/s. (CISCO).

Tabla 8 *Diferencias entre estándares de WI-FI 802.11*

Estándar	Velocidad máxima	Frecuencia	Compatible con modelos anteriores
802.11^a	54 Mb/s	5 GHz	No
802.11b	11 Mb/s	2.4 GHz	No
802.11g	54 Mb/s	2.4 GHz	802.11b
802.11n	600 Mb/s	2.4 GHz o 5 GHz	802.11a/b/g
802.11ac	1.3 Gb/s (1300 Mb/s)	2.4 GHz y 5 GHz	802.11a/n
802.11ad	7 Gb/s (7000 Mb/s)	2.4 GHz, 5 GHz y 60 GHz	802.11a/b/g/n/ac

Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5.

2.6.4.2 Trama inalámbrica 802.11

Las tramas 802.11 incluyen los siguientes campos:

- Campo Versión de protocolo: la versión de la trama 802.11 en uso.
- Campos Tipo y Subtipo: identifican una de las tres funciones y subfunciones de la trama (control, datos y administración).
- Campo A DS: se establece en 1 para las tramas de datos destinadas al sistema de distribución (dispositivos en la estructura inalámbrica).
- Campo Desde DS: se establece en 1 para las tramas de datos que salen del sistema de distribución.
- Campo Más fragmentos: se establece en 1 para las tramas que tienen otro fragmento.

- Campo Reintentar: se establece en 1 si la trama es una retransmisión de una trama anterior.
- Campo Administración de energía: se establece en 1 para indicar que un nodo estará en el modo de ahorro de energía.
- Campo Más datos: se establece en 1 para indicarle a un nodo en el modo de ahorro de energía que se almacenan más tramas en búfer para ese nodo.
- Campo Privacidad equivalente por cable (WEP): se establece en 1 si la trama contiene información encriptada mediante WEP para propósitos de seguridad
- Campo Orden: se establece en 1 en una trama de tipo de datos que utiliza la clase de servicio Estrictamente ordenada (no requiere reordenamiento).
- Campo Duración/ID: según el tipo de trama, representa el tiempo que se requiere en microsegundos para transmitir la trama o una identidad de asociación (AID) para la estación que transmitió la trama.
- Campo Dirección de destino (DA): contiene la dirección MAC del nodo de destino final en la red.
- Campo Dirección de origen (SA): contiene la dirección MAC del nodo que inició la trama.
- Campo Dirección del receptor (RA): contiene la dirección MAC que identifica al dispositivo inalámbrico que es el destinatario inmediato de la trama.
- Campo Número de fragmento: indica el número de cada fragmento de la trama.
- Campo Número de secuencia: indica el número de secuencia asignado a la trama.
Las tramas retransmitidas se identifican con números de secuencia duplicados.
- Campo Dirección del transmisor (TA): contiene la dirección MAC que identifica al dispositivo inalámbrico que transmitió la trama.

- Campo Cuerpo de la trama: contiene la información que se transporta. En las tramas de datos; generalmente se trata de un paquete IP.
- Campo FCS: contiene una comprobación de redundancia cíclica (CRC) de 32 bits de la trama. (CISCO).

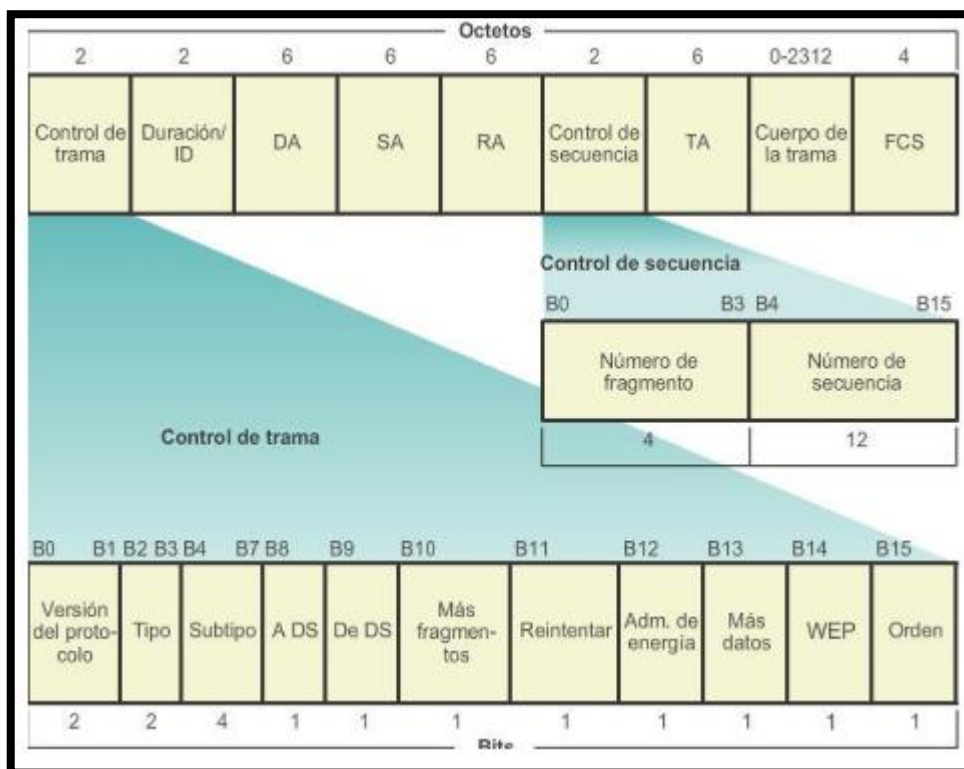


Figura 21 Trama Protocolo LAN inalámbrico 802.11
Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5

2.6.5 Ethernet

Ethernet es la tecnología de red de área local (LAN) de uso más frecuente y predominante. Esta tecnología de red se define en los estándares IEEE 802.2 y 802.3.

Ethernet es la tecnología LAN que admite anchos de banda de datos de 10 Mbps, 100 Mbps, 1 Gbps o 10 Gbps.

Acceso múltiple por detección de portadora con detección de colisiones (CSMA/CD) es el método que utiliza la tecnología Ethernet, donde el dispositivo final supervisa los medios para detectar la presencia de una señal de datos. Si no hay una señal de datos y, si los medios están libres, el dispositivo transmite los datos. Si luego se detectan señales que muestran que otro dispositivo estaba transmitiendo al mismo tiempo, todos los dispositivos dejan de enviar e intentan después. (CISCO).

2.6.5.1 Trama Ethernet

Las tramas Ethernet incluyen los siguientes campos:

- Preámbulo: es utilizado para la sincronización. Además contiene un delimitador para marcar el final de la información de temporización.
- Dirección de Destino: dirección MAC de 6 bytes para el nodo de destino.
- Dirección de Origen: dirección MAC de 6 bytes para el nodo de origen.
- Tipo: valor que indica el protocolo de capa superior que recibirá los datos una vez que finalice el proceso Ethernet.
- Datos: es el PDU, un paquete IPV4, que se transportará a través de los medios.
- Secuencia de Verificación de trama (FCS): valor utilizado para verificar si hay tramas con errores. (CISCO).

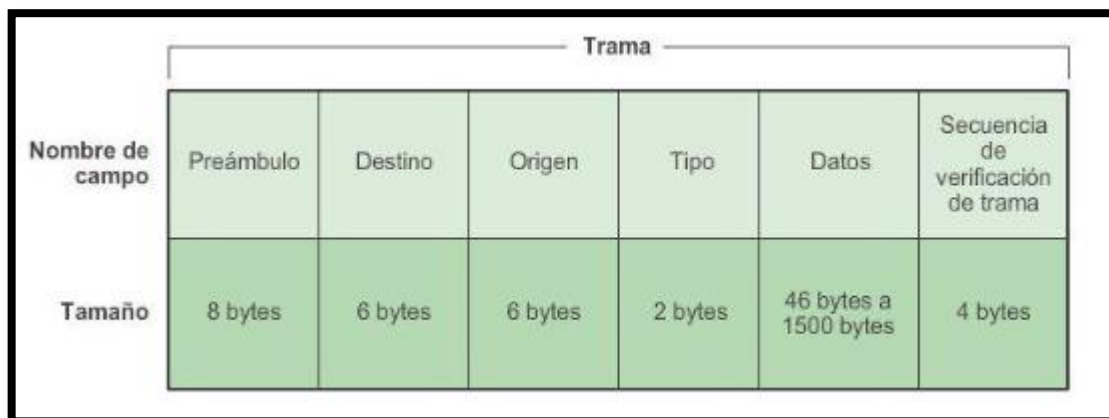


Figura 22 Trama Ethernet

Fuente: CISCO. Principios básicos de enrutamiento y switching, CCNA1 V5

2.7 ESTUDIO DE LA TECNOLOGÍA RFID

2.7.1 Introducción

La identificación por radiofrecuencia (RFID) es una tecnología de captura e identificación automática de información contenida en etiquetas electrónicas (tags). Cuando estas etiquetas entran en el área de cobertura de un lector RFID, éste envía una señal para que la etiqueta le transmita la información almacenada en su memoria, habitualmente un código de identificación. (AETIC). La tecnología RFID recupera la información contenida en la tag o etiqueta vía radiofrecuencia y no requiere de que exista contacto físico entre el lector RFID y la tag o etiqueta, aunque en la mayoría de los casos se requiere una cierta proximidad de esos elementos. En un sistema RFID el elemento a identificar puede ser un objeto, persona o animal.

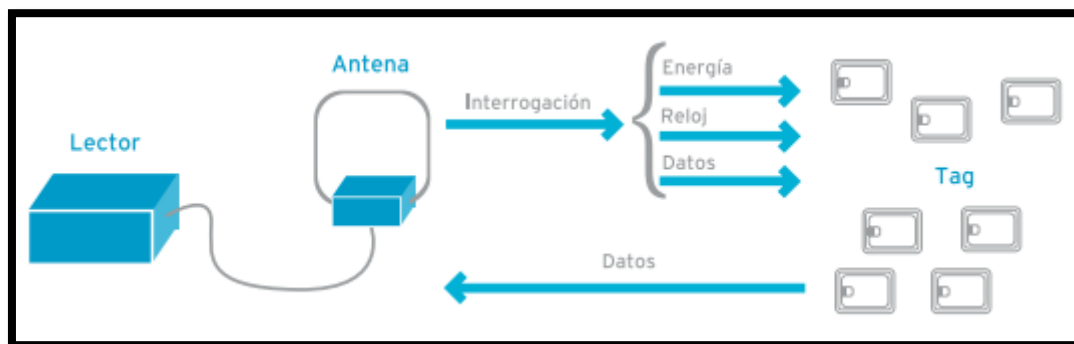


Figura 23 Esquema general de funcionamiento de un sistema RFID
Fuente: AETIC, O. Y. La tecnología RFID: Usos y oportunidades

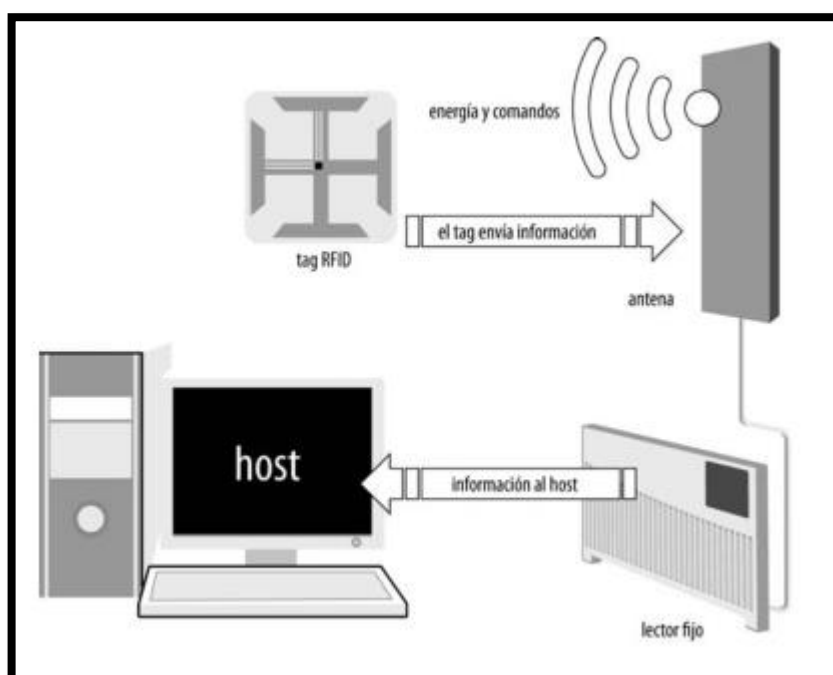


Figura 24 Arquitectura Básica tecnología RFID
Fuente: TELETRONICA. Introducción a la identificación por radio frecuencia (Primera. ed.)

2.7.2 Componentes del Sistema

Los componentes básicos de un sistema RFID son:

- Etiquetas RFID o Tag.
- Antenas.
- Lector RFID.

2.7.2.1 Etiquetas RFID o Tag

La etiqueta RFID se pueden presentar de varias formas: tarjeta, pulsera, botón, etiqueta adhesiva, disco plástico, chapa PVC, ampolla de vidrio, cilindro de plástico, llaveros, o incluso puede ser inyectado bajo la piel.



Figura 25 Ejemplos de tags de diferentes formas y tamaños
Fuente: AETIC, O. Y. La tecnología RFID: Usos y oportunidades

2.7.2.1.1 Composición de una etiqueta RFID

Una etiqueta está compuesta por:

- **Chip o circuito integrado (1):** Almacena la información ejecuta los comandos específicos.
- **Antena (2):** Una bobina que funciona como antena, emite y/o recibe las ondas de radio con la información contenida en el chip.
- **Sustrato (3):** Material que mantiene el chip y la antena juntos y protegidos. Además, es el sustrato el que determina la forma física del tag. (ARMENDARIZ, 2009).

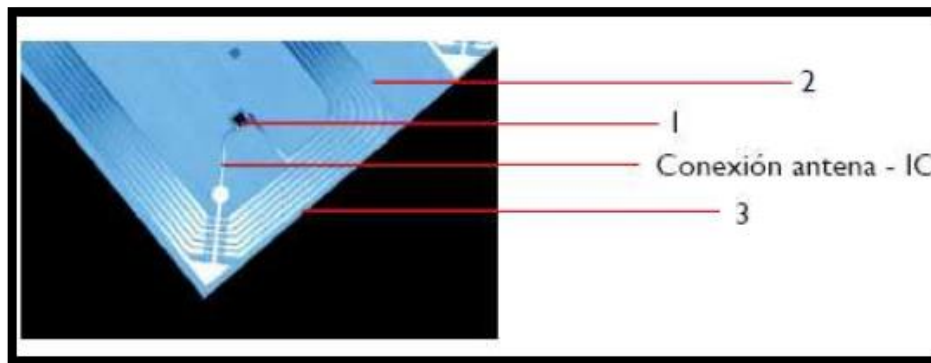


Figura 26 Estructura Básica de un Tag RFID

Fuente: ARMENDARIZ, J. C. (2009). Análisis comparativo de las tecnologías RFID, HID y AWID para proponer un sistema para la ubicación y seguridad de los libros en la biblioteca de la FIEE de la E.P.N. Escuela Politécnica Nacional, Quito-Ecuador

2.7.2.1.2 Tipos de Etiquetas RFID o Tag

Las etiquetas de identificación por frecuencia de radio pueden ser pasivas o activas. Las etiquetas pasivas se activan sólo cuando se encuentran en el campo de transmisión de frecuencia de radio de un lector, mientras que las etiquetas activas funcionan con baterías y emiten constantemente una señal de frecuencia de radio. (MATURANA, 2006).

A continuación se presenta dos tablas, la Tabla 9 representa algunas diferencias técnicas entre las etiquetas RFID activas y pasivas.

Tabla 9 Diferencias técnicas entre tecnologías RFID activas y pasivas

	RFID ACTIVO	RFID PASIVO
Fuente poder etiqueta	En el interior de la etiqueta	Energía transferida desde el lector RFID vía radiofrecuencia
Presencia batería en etiqueta	SI	NO
Disponibilidad energía TAG	Continua	Solo cuando es incluido desde el lector

Energía señal transmitida desde el lector al TAG-	BAJA	ALTA (Debe incluir al TAG)
Energía señal transmitida desde el TAG hacia el lector RFID	ALTA	BAJA
Calidad de información transmitida	ALTA	BAJA

Fuente: ARMENDARIZ, J. C. (2009). Análisis comparativo de las tecnologías RFID, HID y AWID para proponer un sistema para la ubicación y seguridad de los libros en la biblioteca de la FIEE de la E.P.N. Escuela Politécnica Nacional, Quito-Ecuador.

Mientras que en la tabla 10 se presentan las capacidades funcionales de las etiquetas RFID activas y pasivas.

Tabla 10 *Capacidades funcionales de tecnologías RFID activos y pasivos*

	RFID ACTIVO	RFID PASIVO
Rango Comunicación	Alcance mayor (100m o más)	Pequeño y muy pequeño alcance (3m o menor)
Conexión multitag	Puede recoger información de 1000 tags en reposo con un lector simple.	Recoge información de cientos de tags con un lector simple
	Lee información de 20 tags moviéndose a velocidades de 100 mph.	Lee información de 20 tags moviéndose a menos de 3mph.
Capacidad sensores	Habilidad de monitoreo continuo y grabación de entradas de sensores: datos y tiempo (hora, fecha).	Habilidad de leer y transferir calores de sensores solo cuando es inducido por el lector; no guardar fecha ni hora.
Transferencia datos	Lee y escribe datos (EJ. 128 KB) con sofisticados algoritmos de búsqueda y capacidad de acceso.	Bajo nivel de almacenamiento de información (Ej. 128 bytes)

Fuente: ARMENDARIZ, J. C. (2009). Análisis comparativo de las tecnologías RFID, HID y AWID para proponer un sistema para la ubicación y seguridad de los libros en la biblioteca de la FIEE de la E.P.N. Escuela Politécnica Nacional, Quito-Ecuador.

Y en la tabla 11 se presenta una comparativa entre las tags pasivas y las tags activas.

Tabla 11 *Comparativa entre características de tags pasivos y tags activos*

Tag Pasivo	Tag Activo
Funciona sin batería	Funciona con batería
Relativamente económico	Relativamente costoso
Ciclo de vida ilimitado	Ciclo de vida limitado por la batería
Poco peso	Mayor peso
Alcance limitado (3 - 5m)	Mayor alcance (100m)
Sensibilidad al ruido	Mayor inmunidad ante presencia de ruido
Dependencia de la señal del dispositivo lector	Transmisor propio
Requiere dispositivos lectores potentes	Relaja el requisito de potencia de los lectores
Velocidad de transmisión baja	Velocidad de transmisión alta
Lectura simultánea baja	Lectura simultánea alta
Alta sensibilidad de orientación	Menor sensibilidad de orientación

Fuente: AETIC, O. Y. La tecnología RFID: Usos y oportunidades.

2.7.2.2 Antenas

Dispositivo que utiliza las ondas de radio para leer y escribir datos en los Tags. Existen de diferentes tamaños y formas según la frecuencia de operación de la onda de la señal.

La función principal es enviar una señal hacia las etiquetas y recibir el resultado de identificación (ID), pero existe también una función secundaria importante que realizan las antenas y es el de inducir un pequeño voltaje en las etiquetas pasivas para su correcta polarización.

Su funcionalidad depende del tipo de Tag a usar; es decir si se utiliza una etiqueta de radio frecuencia en modalidad activa o pasiva. La capa física RFID consiste en la forma

en que el radio y las antenas permiten la comunicación mutua y la transferencia de información. Existen dos tipos de antenas: antenas móviles y fijas. (ARMENDARIZ, 2009).

2.7.2.2.1 Características de las antenas

Hay tres características de las antenas que contribuyen a la legibilidad de la etiqueta.

- **Patrón:** Se refiere al campo de energía tridimensional creado por la antena. Esto es también conocido como el área de lectura.
- **Ganancia y atenuación:** La ganancia de la antena de un lector es fijada en relación a las regulaciones gubernamentales. No obstante, la señal puede reducirse o atenuarse para limitar el rango de lectura de la etiqueta o para dirigirla sólo a las etiquetas que uno desea leer.
- **Polarización:** Se refiere a la orientación de la transmisión del campo electromagnético. (TELECTRONICA).

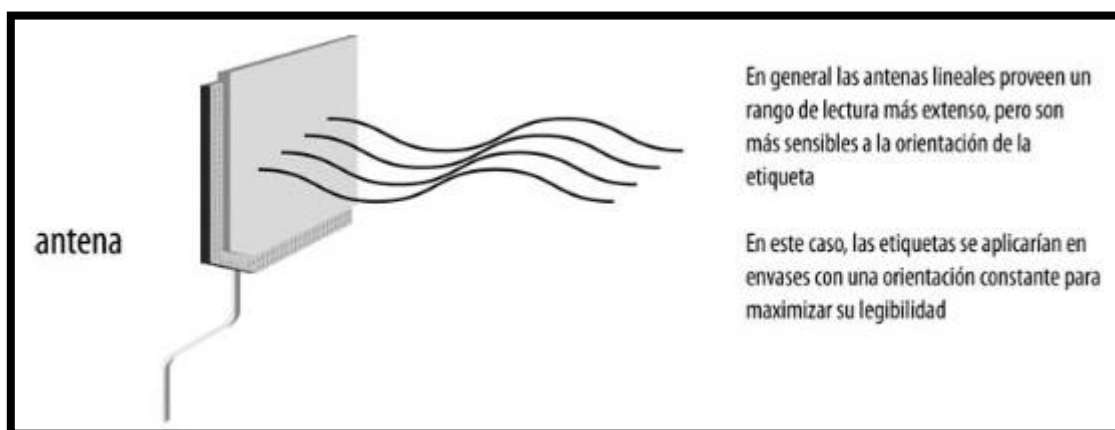


Figura 27 Polarización Lineal

Fuente: TELECTRONICA. Introducción a la identificación por radio frecuencia (Primera. ed.)

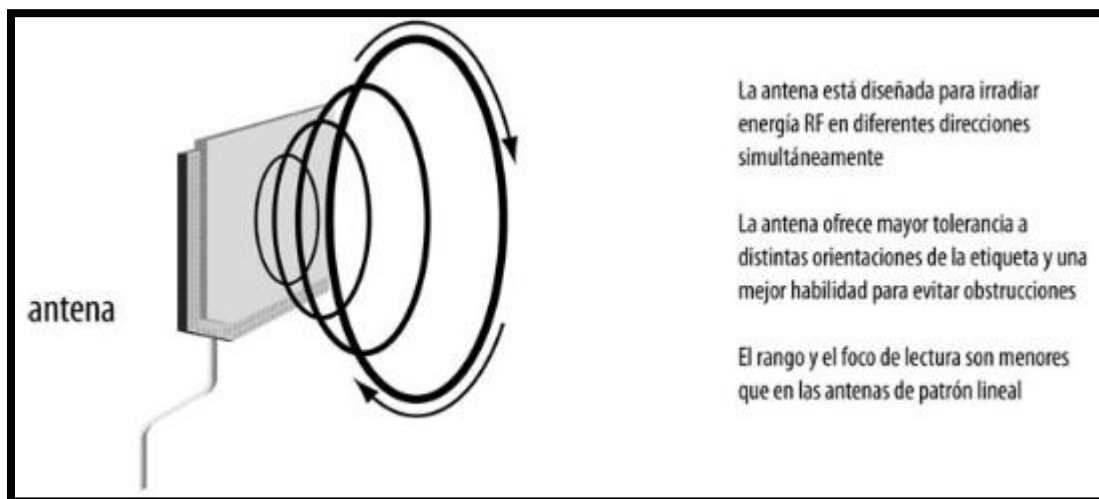


Figura 28 Polarización Circular

Fuente: TELECTRONICA. Introducción a la identificación por radio frecuencia (Primera. ed.)

2.7.2.3 Lector RFID

Es el dispositivo que se encarga del procesamiento de la información. El sistema de control, búsqueda o almacenamiento de información se enlaza con el receptor para la obtención de la identificación (ID) o información necesaria. (ARMENDARIZ, 2009). En nuestro caso el lector RFID se enlaza con nuestro sistema de control que es la plataforma de Arduino, y puede comunicarse de diferentes maneras: comunicación serial, paralela, comunicación SPI, USB o algún interfaz inalámbrico.

2.7.2.3.1 Operación básica de un lector

Para la mayoría de las aplicaciones, los lectores operarán automáticamente o como equipos dirigidos. Los lectores utilizan la banda de 902-928 MHz (USA) dividida en aproximadamente 60 canales. La metodología de modulación es denominada Frequency Hopping Spread Spectrum y ha sido establecida por la FCC para minimizar la interferencia

con otros dispositivos de RF. Algoritmos anti-colisión son utilizados para leer y clasificar ingresos múltiples y simultáneos de distintos tags. (TELECTRONICA).

- **Lectura Fija:** Un lector puede setearse para operar en forma constante realizando lecturas fijas y acumulando listas de tags en su memoria.
- **Modo Directo/Interactivo:** Los lectores que operan bajo esta modalidad responderán a los comandos del host.

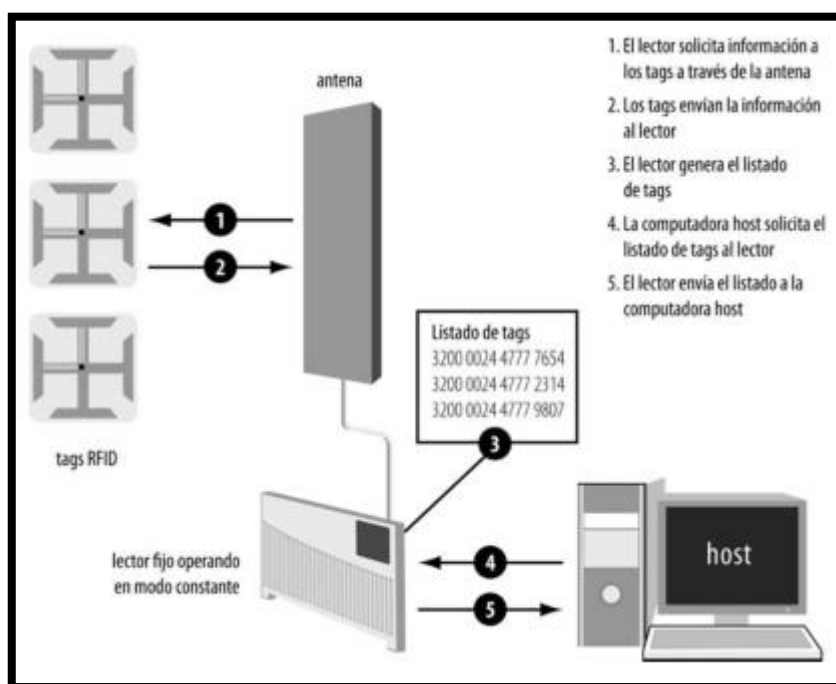


Figura 29 Lector fijo seteado para leer etiquetas automáticamente
Fuente: TELECTRONICA. Introducción a la identificación por radio frecuencia (Primera. ed.)

2.7.3 FRECUENCIAS DE OPERACIÓN

Los diferentes sistemas de RFID utilizan las siguientes bandas de frecuencias que actualmente están en el mercado:

Tabla 12 *Bandas de frecuencias utilizadas en RFID*

Bandeja de Frecuencias	Descripción	Rango
125 kHz - 134 kHz	LF (Baja Frecuencia)	Hasta 45 cm.
13,553 MHz - 13,567 MHz	HF (Alta Frecuencia)	De 1 a 3 m.
400 MHz - 1.000 MHz	UHF (Ultra Alta Frecuencia)	De 3 a 10 m.
2,45 GHz - 5,4 GHz	Microondas	Más de 10 m.

Fuente: AETIC, O. Y. La tecnología RFID: Usos y oportunidades.

Para seleccionar la óptima frecuencia de radio de operación se requiere el estudio de varios factores, incluyendo el rendimiento, factores regulatorios y coexistencia con otras tecnologías inalámbricas. (AETIC).

A continuación se detalla en la tabla 13 las características principales y ejemplos de aplicación de diferentes sistemas de RFID según la banda de frecuencia que utilizan:

Tabla 13 *Características principales y ejemplos de aplicación de las distintas bandas de frecuencia utilizadas en RFID*

Banda de Frecuencias	Características del sistema	Ejemplos de Aplicaciones
LF (de 100 a 500 kHz). Típico 125 a 134 kHz. Internacional	Corto alcance. Poca velocidad de transmisión, Relativamente económico, Gran penetración en los materiales (líquidos), Trabaja bien junto a metales.	Control de acceso, Identificación de animales, Control de inventario, EAS (Antirrobo), Llaves de automóvil
HF. Típico 13,56 MHz. Internacional	Corto/medio alcance, Velocidad de transmisión media, Puede leer a través de líquidos y en entornos húmedos, Problemático junto a metales, Moderadamente caro, Posibilidad de actuar como lector o etiqueta en función del escenario de utilización (NFC)	Control de acceso, Tarjetas inteligentes, EAS (Antirrobo), Inventario en bibliotecas, Gestión de almacén, Control de equipajes, Gestión de lavandería, Identificación de pacientes, Pago con el móvil y captura de datos con solo acercar el móvil (NFC)

UHF (de 400 a 1.000 MHz) Típico 850 - 950 MHz	Largo alcance, Alta velocidad de transacción, Mecanismos de anticolisión, Problemático con líquidos y metales, Problemático en entornos húmedos, En metal genera interferencias, Moderadamente caro	Gestión de artículos, Gestión de la cadena de suministro, Gestión de almacén, Gestión de expediciones, Trazabilidad
Microondas (de 2,4 a 6 GHz)	Medio alcance, Características similares a los tags UHF pero con mayor velocidad de transmisión, Mayor precio.	Control ferroviario, Peajes de autopista, Localización

Fuente: AETIC, O. Y. La tecnología RFID: Usos y oportunidades.

2.8 GENERALIDADES DE LOS SERVIDORES

2.8.1 Modelo cliente-servidor

Los protocolos de capa aplicación interactúan con aplicaciones de usuarios finales, en este modelo el dispositivo que solicita la información se denomina cliente, y el dispositivo que responde a solicitudes se denomina servidor.

El cliente empieza el intercambio de datos solicitándole datos al servidor, el servidor responde al cliente enviando uno o más streams de datos. Además del intercambio también puede requerir la autenticación del usuario y la identificación de un archivo de datos que se vaya a transferir.

La transferencia de datos de un cliente a un servidor se conoce como subida y la transferencia de datos de un servidor a un cliente se conoce como descarga. (CISCO).

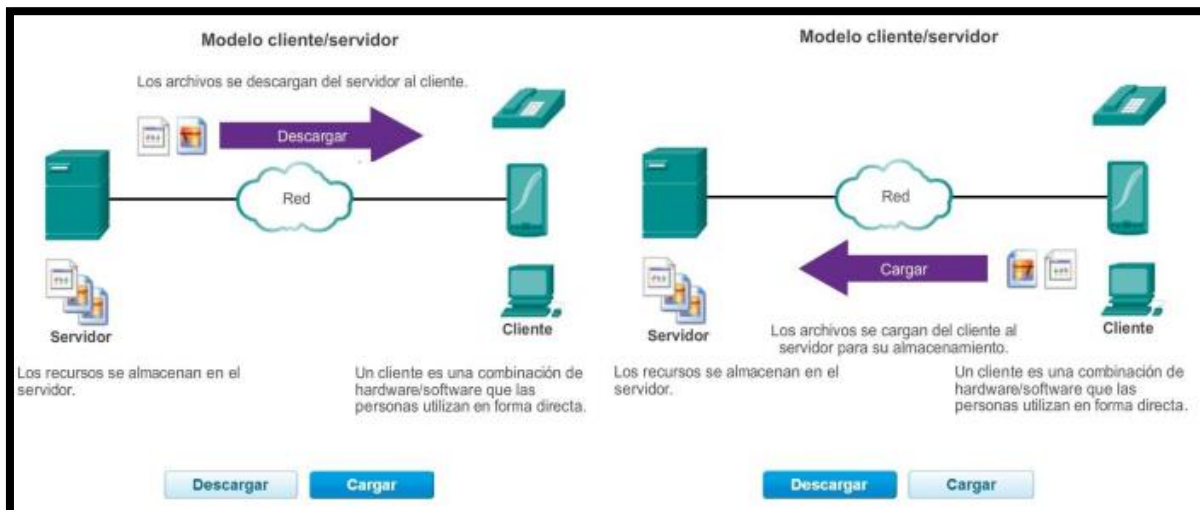


Figura 30 Modelo cliente-servidor
 Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5

2.8.2 Protocolos de capa aplicación

Existen muchos protocolos de capa aplicación, a continuación se detallan algunos protocolos de capa aplicación que forman parte del trabajo:

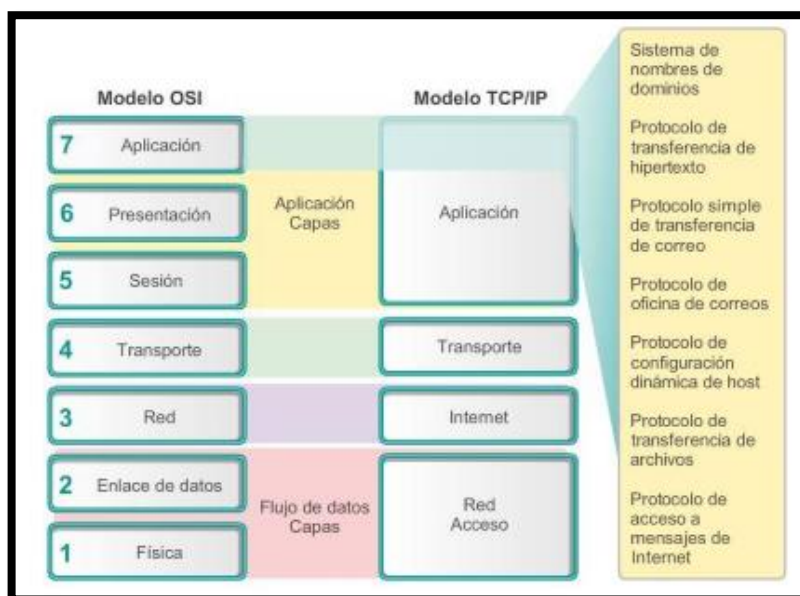


Figura 31 Protocolos de capa Aplicación del modelo OSI y TCP/IP
 Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5

2.8.2.1 Servidor FTP

El servidor FTP (Protocolo de transferencia de archivos) es otro protocolo que trabaja de capa de aplicación en el modelo cliente-servidor. (CISCO). Los programas FTP permiten a los usuarios transferir archivos a un sistema remoto, ejecutando un programa de servidor FTP y desde este. Cualquier sistema Linux opera como servidor FTP, solo debe ejecutarse como software de servidor un daemon FTP con la configuración apropiada (PETERSEN, 2009).

Para transferir datos correctamente, FTP necesita dos conexiones entre el cliente y servidor, la primera conexión para los comandos y las respuestas, y la otra conexión para la transferencia de archivos.(PETERSEN, 2009).

“El cliente establece la primera conexión al servidor para el tráfico de control, que está constituido por comandos del cliente y respuestas del servidor.” (CISCO).

“El cliente establece la segunda conexión al servidor para la transferencia de datos propiamente dicha. Esta conexión se crea cada vez que hay datos para transferir.” (CISCO).

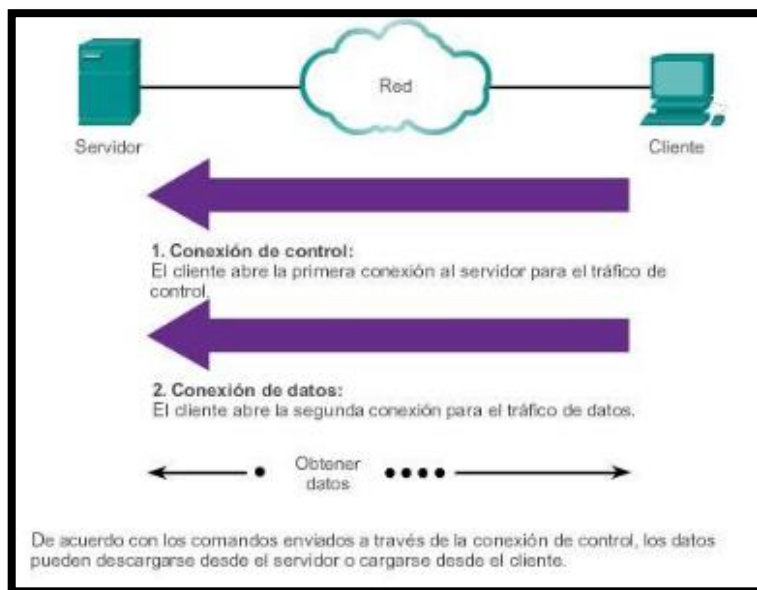


Figura 32 Proceso FTP

Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5

Tabla 14 Servidores FTP más comunes

Servidor FTP	Sitio
Very Secure FTP Server (vsftpd)	vsftpd.beasts.org
ProFTPD	proftpd.org
PureFTP	pureftpd.org
NcFTPD	ncftpd.org
SSH sftp	openssh.org
Washington University web server (WU-FTPD)	wu-ftpd.org
Tux	Web server con Capacidad FTP
Gssftpd	Servidor FTP Kerberos

Fuente: PETERSEN, R. (2009). LINUX Manual de Referencia (Sexta ed.): Mc Graw Hill.

2.8.2.2 Servidor SSH

El protocolo de Shell seguro (SSH) facilita las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor y que permite a los usuarios conectarse a un host remotamente. (PETERSEN, 2009). El protocolo SSH proporciona autenticación de contraseña y usa encriptación cuando transporta datos de la sesión. Así logra mantener en

privado la ID del usuario, la contraseña y los detalles de la sesión de administración. (CISCO).

Programas de emulación de terminal

Existen varios programas disponibles para conectarse a un dispositivo de red.

Algunos de estos programas son: Putty, Tera Term, SecureCRT, Hyperterminal, ZOC.

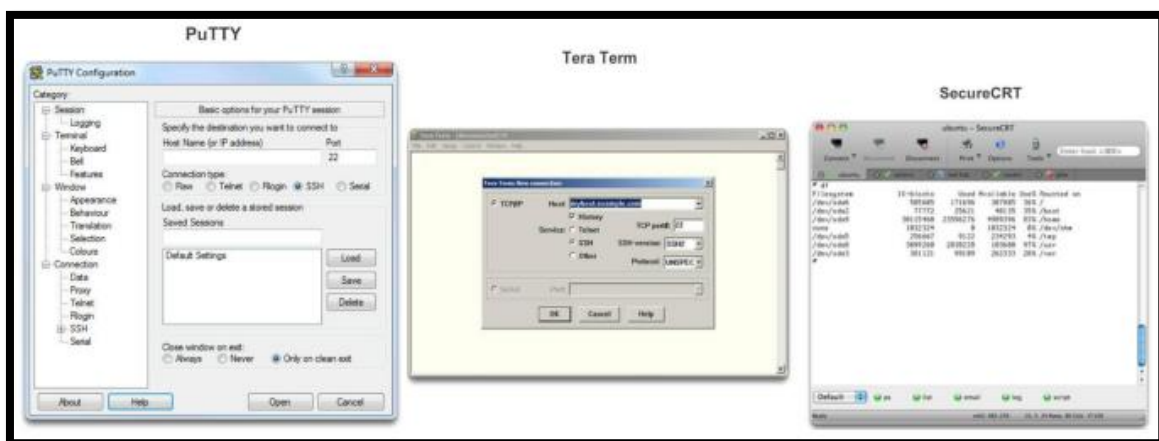


Figura 33 Programas de emulación de terminal

Fuente: CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5

2.8.2.3 Servidor WEB

Cuando en un explorador web se escribe una dirección web o URL (Localizador uniforme de recursos), el explorador establece una conexión con el servicio Web que se ejecuta en el servidor mediante el protocolo HTTP. (CISCO).

A través de los exploradores web una PC puede conectarse a la World Wide Web y acceder a los recursos almacenados en un servidor Web. Los clientes Web establecen

conexiones al servidor y solicitan recursos deseados. El servidor responde con el recurso y, al recibirlo, el explorador interpreta los datos y los presenta al usuario. (CISCO).

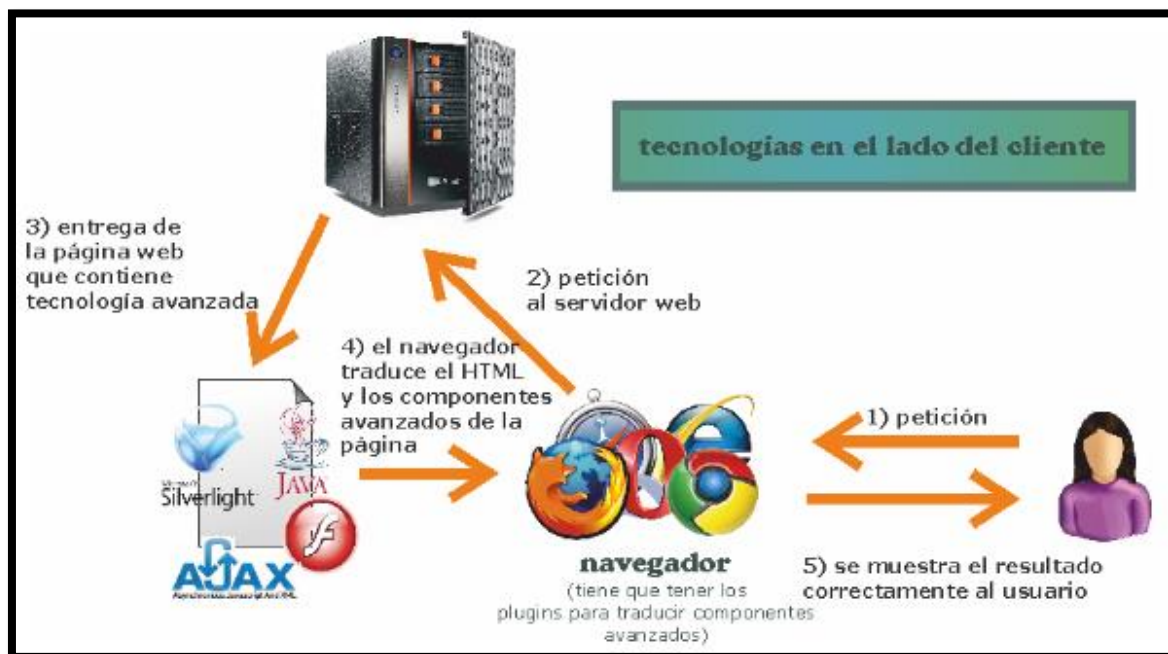


Figura 34 Funcionamiento de las tecnologías ricas de Internet usando tecnologías en el lado del cliente
Fuente: SANCHEZ, J. (2012). Servidores de Aplicaciones Web

HTTP es un protocolo de solicitud/respuesta, y especifica tres tipos de mensajes para la comunicación: GET, POST y PUT.

GET.- solicitud de datos que realiza el cliente. El explorador envía el mensaje GET al servidor Web para solicitar las paginas HTML. Cuando el servidor recibe la solicitud GET, el servidor responde con una línea de estado y un mensaje propio. El mensaje puede contener un mensaje de error o de información. (CISCO).

POST y PUT.- ambos son utilizados para subir datos al servidor Web, el mensaje POST introduce datos en un formulario por ejemplo, y el mensaje PUT carga recursos o contenido como imágenes al servidor Web. (CISCO).

En la siguiente figura 35, se puede observar el funcionamiento cuando se crean páginas que contienen componentes que pertenecen a tecnologías que deben ser interpretadas en el servidor. Es decir, cuando un usuario pide una página web, el servidor web se da cuenta que contiene elementos especiales pide a un servidor de aplicaciones como apache que traduzca esos elementos. El servidor de aplicaciones dará el resultado al servidor web en un formato asumible por el (normalmente en el propio HTML) y este finalmente continua el proceso habitual entregando la página al navegador. (SANCHEZ, 2012).

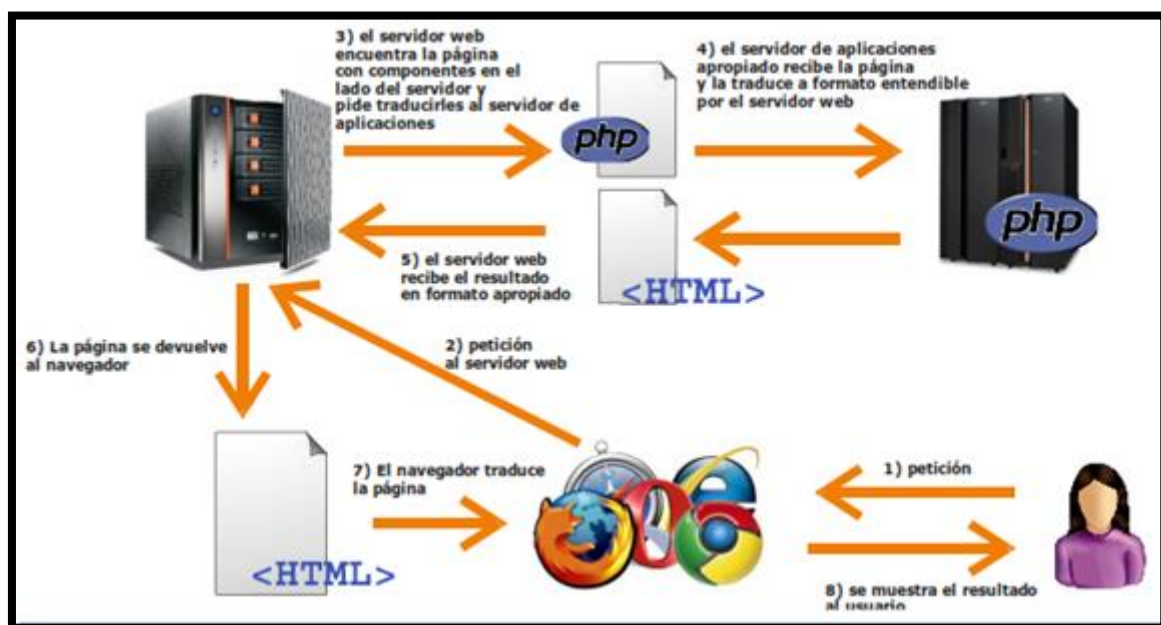


Figura 35 Funcionamiento de las aplicaciones creadas para ser interpretadas en el lado del servidor
Fuente: SANCHEZ, J. (2012). Servidores de Aplicaciones Web

2.8.2.4 Servidor LAMP

LAMP son las siglas de Linux + Apache + MySQL + PHP/Perl/Python, y se le conoce como una plataforma para desarrollo y ejecución de aplicaciones web. Todos los elementos que forman LAMP son software libre.

Una plataforma Web tiene cuatro componentes básicos: un sistema operativo, un servidor Web, una base de datos y un lenguaje de programación. (BAQUE, 2011).



Figura 36 Servidor LAMP

Fuente: <http://recursos.cepindalo.es/mod/book/view.php?id=578&chapterid=188>

La “L” significa el sistema operativo Linux. Este sistema operativo en la actualidad se ha posicionado fuertemente en el sector de los servidores.

La “A” denota del servidor web Apache, es el encargado de servir las páginas web hacia quien lo solicita. Apache es el servidor web HTTP de código abierto, potente y muy estable usado en la mayoría de los servidores web de Internet.

La “M” proviene de MySQL, es un sistema de gestión de Bases de Datos que usa el modelo relacional con SQL, su manejo es sencillo y de alto desempeño.

La “P” proviene de los lenguajes de programación de código abierto PHP, Perl y Python. (BAQUE, 2011).

PHPmyAdmin: Sirve para administrar la base de datos a través de interfaz web.

CAPÍTULO III

Descripción General

Este capítulo se basa en el diseño y fabricación del sistema, que permite registrar automáticamente el nombre, la fecha, hora y cantidad de leche producida por cada bovino. Se diseñará diagramas de bloques del sistema, algoritmos de programación, y el diseño de la etapa de control del ordeño mecánico. Así, como también su respectiva recolección de información en la base de datos para los servidores.

3.1 CONSIDERACIONES DEL DISEÑO

Se ha tomado en cuenta aspectos fundamentales para que el sistema funcione correctamente, para lo cual se debe considerar:

- Enlace Inalámbrico.- para que se pueda acceder a la base de datos, al servidor web y para que se puedan enviar los correos electrónicos, es necesario del recurso del Internet. Para lo cual es necesario realizar un enlace inalámbrico entre la estación de ordeño y el Rack de comunicaciones, el cual está situado en el laboratorio de la Granja Experimental La Pradera.
- Energía eléctrica.- es un requerimiento importante para el funcionamiento de todo el sistema, este subsistema de energía eléctrica debe garantizar el abastecimiento de energía necesaria para la operación normal de todo el sistema.

- Protección de sistemas eléctricos.- para prevenir que sobrecargas afecten el funcionamiento de los dispositivos de propósito general, es fundamental la presencia de elementos de protección como breakers, fusibles.

3.2 DESCRIPCIÓN DEL FUNCIONAMIENTO DEL SISTEMA

Este apartado consiste en dividir por etapas todo el funcionamiento del sistema, para entender de mejor manera el desarrollo de todo el sistema.

En cada etapa se describen los elementos que intervienen, los recursos que se utilizan de los elementos y la acción que realizan cada uno de ellos.

3.2.1 Primera Etapa

La primera etapa se basa en la identificación electrónica del animal, en la cual los elementos que intervienen son:

- El lector RFID.
- La etiqueta RFID.
- EL microcontrolador.
- El operador o persona encargada.

3.2.1.1 Recursos

Los recursos que se utilizan del microcontrolador y del lector RFID en esta etapa son:

- La comunicación SPI por la cual se comunican los lectores RFID con el microcontrolador.

3.2.1.2 Acción

El operador tiene que acercar la etiqueta de cada bovino (etiqueta RFID incluida en el arete de cada animal) al lector RFID, el lector transmitirá la identificación de cada bovino al microcontrolador. El microcontrolador será el encargado de procesar estos datos, y en la pantalla GLCD se visualizará a través de un mensaje si la etiqueta fue identificada correctamente o no.

Revisar en ANEXO 01 el script de configuración y la utilización de las respectivas librerías.

3.2.2 Segunda Etapa

La segunda etapa consta de la activación del sistema y la medición de la cantidad de leche obtenida, los elementos que intervienen en esta etapa son:

- Pulsador.
- Sensores (Sensor flujo).
- Actuadores (led, relés, transistores, válvula solenoide, motor eléctrico).
- El Microcontrolador.
- El operador o persona encargada.

3.2.2.1 Recursos

Los recursos que se utilizan del microcontrolador en esta etapa son:

- Las entradas digitales del microcontrolador para conectar el pulsador, para la activación del sistema.
- Las salidas digitales del microcontrolador para conectar los leds, que indican el estado de la activación del sistema, de la válvula solenoide y del motor eléctrico.
- Se utilizará una función especial de interrupción del microcontrolador, para conectar con el sensor de flujo.

3.2.2.2 Acción

El operador o la persona encargada, tienen que presionar el pulsador que se encuentra conectado al microcontrolador, para activar el sistema. El microcontrolador activará las salidas digitales para visualizar a través de un led, que el sistema se encuentra activo, y que el motor eléctrico fue activado en un sentido para colocar las pezoneras. El operador tendrá un tiempo determinado, para proceder a colocar las pezoneras en la ubre del animal. Después de ese tiempo se activará otra salida digital para visualizar a través de un led, que la válvula solenoide se ha activado y permitirá el paso del flujo de leche por el conducto, para luego por medio del sensor de flujo obtener la cantidad de leche producida. En la pantalla GLCD se podrá visualizar el tiempo determinado, el flujo del caudal de la leche que está siendo medido por el sensor de flujo, y cuando el proceso haya finalizado se visualizará la cantidad de leche total producida por cada bovino.

Para que la medición de la leche sea lo más precisa, se toma muestras del flujo del caudal cada segundo y se realiza una sumatoria. Cuando el sensor de flujo ya no tiene nada que sensar, se realiza un promedio del flujo del caudal de la leche (L/min) en el tiempo transcurrido, y luego se realiza un proceso de conversión para obtener los litros de leche producidos por cada bovino.

Revisar en ANEXO 01 el script de configuración y la utilización de las respectivas librerías.

3.2.3 Tercera Etapa

La tercera etapa se basa en el control del sistema y en la comunicación del microcontrolador con la minicomputadora, los elementos que intervienen en esta etapa son:

- Sensores (Sensor flujo).
- Actuador (relé, motor eléctrico, válvula solenoide, led, buzzer).
- El Microcontrolador.
- La minicomputadora.
- Módulo RTC (Reloj de tiempo real).

3.2.3.1 Recursos

Los recursos que se utilizan del microcontrolador en esta etapa son:

- Se utilizará una función especial de interrupción del microcontrolador, para conectar con el sensor de flujo.

- Las salidas digitales del microcontrolador para encender el led, que indica que el proceso de ordeño ha terminado completamente; y otra salida digital para activar en sentido contrario el motor eléctrico, para que realice el retiro automático de las pezoneras.
- La comunicación serial (UART) para conectar con la minicomputadora.

Los recursos que se utilizan de la minicomputadora en esta etapa son:

- La comunicación serial (UART) de la minicomputadora, para conectar con el microcontrolador.
- La comunicación I2C para conectar el módulo RTC (Reloj de tiempo real), y obtener a través de este módulo la fecha y hora en tiempo real, sin la necesidad de un servidor NTP (Protocolo de tiempo de Red).
- Los pines GPIO (General Purpose Input/Output) donde se conecta un pulsador, como entrada para apagar todo el sistema correctamente; ya que si se desconecta todo el sistema eléctrico, la minicomputadora puede sufrir múltiples daños.

3.2.3.2 Acción

El microcontrolador y la minicomputadora se conectarán a través de la comunicación serial (UART) de ambos dispositivos respectivamente, el microcontrolador será el encargado de obtener la cantidad de leche producida por el sensor de flujo y enviar a través del bus de comunicación serial los datos obtenidos a la minicomputadora. El sensor de flujo cuando ya no tenga nada que sensor, se interrumpirá el funcionamiento a través de la válvula solenoide para impedir el paso de presión de aire proveniente del ordenó mecánico, y consecuentemente se retirarán automáticamente las pezoneras del animal a

través del motor eléctrico. El microcontrolador para evidenciar al operador que el bovino se ha ordeñado completamente, activará unas salidas digitales para visualizar a través de un led o de una señal sonora, que el proceso ha terminado y que el operador puede continuar con el siguiente ordeño.

La fecha y la hora son obtenidas a través del módulo RTC en tiempo real, cuando el microcontrolador envía la cantidad de leche producida por cada bovino hacia la minicomputadora. Esto se lo realiza mediante el lenguaje de programación Python en un script.

Revisar en ANEXO 01 Y ANEXO 02 el script de configuración y la utilización de las respectivas librerías.

3.2.4 Cuarta Etapa

La cuarta etapa se basa en el almacenamiento de los datos obtenidos, en la base de datos, en un archivo de texto; y en habilitar un servidor Web, que permita gestionar la base de datos. Los elementos que intervienen en esta etapa:

- La minicomputadora.

3.2.4.1 Recursos

Los recursos que se utilizan de la minicomputadora en esta etapa son:

- Servidor LAMP (Linux, Apache, MySQL, PHP, Python)

- phpMyAdmin permite gestionar la base de datos a través de una interfaz web.

3.2.4.2 Acción

En la minicomputadora se procederá a crear una base de datos (MySQL), para almacenar el nombre, fecha, hora, datos obtenidos de la producción de leche del animal, el número de ordeño y un ID que permite identificar de manera única cada operación. Estos datos son obtenidos, mediante el lenguaje de programación Python en un script, para lo cual es necesario la instalación de algunas librerías, como import serial para la comunicación serial entre la minicomputadora y el microcontrolador, la librería import MySQLdb para la administración de la base de datos, la librería import time e import datetime para la obtención de la fecha y hora; también es fundamental una librería denominada os que me permite ejecutar comandos del sistema de la minicomputadora para el apagado de todo el sistema (shutdown -h now), y el envío del correo electrónico. Para gestionar la base de datos a través de una interfaz web se hace uso de la herramienta phpMyAdmin. Para que los datos obtenidos, se han almacenados en un archivo de texto, se hace uso de objetos en el lenguaje de programación de Python.

Una vez creada la base de datos, se procederá habilitar un servidor Web (Apache), que permita gestionar la base de datos obtenida, y así como consultar resultados previos del ganado bovino a través de una interfaz gráfica. El lenguaje de programación de script web que se utilizará es PHP y HTML.

Revisar en el Manual del Administrador la instalación del servidor LAMP y sus respectivas librerías.

Revisar en ANEXO 02 el script de configuración y la utilización de las respectivas librerías.

3.2.5 Quinta Etapa

Como quinta etapa se enviará un correo electrónico, al administrador o persona encargada de la granja, que contenga un archivo con la información diaria de cada bovino.

Los elementos que intervienen en esta etapa son:

- La minicomputadora.

3.2.5.1 Recursos

Los recursos que se utilizan de la minicomputadora en esta etapa son:

- Servicio de correo electrónico de Google (Gmail).

3.2.5.2 Acción

Esta etapa la realizará la minicomputadora, y se utilizará como remitente el servicio de correo electrónico de Google (Gmail). El archivo de texto donde se encuentran almacenados los datos obtenidos, se envían mediante objetos del lenguaje de programación de Python en un script, para lo cual es necesario una librería denominada os, que me permite

ejecutar comandos del sistema de la minicomputadora para el envío del correo electrónico. Para esto es necesario instalar un paquete denominado mpack, el cual me permite enviar archivos de textos a un correo electrónico.

Revisar en ANEXO 02 el script de configuración y la utilización de las respectivas librerías.

3.2.6 Sexta Etapa

Como sexta etapa se creará un servidor FTP. Los elementos que intervienen en esta etapa son:

- La minicomputadora.

3.2.6.1 Recursos

Los recursos que se utilizan de la minicomputadora en esta etapa son:

- Servidor FTP (vsftpd).
- Cliente FTP (WinSCP).

3.2.6.2 Acción

Esta etapa la realizará la minicomputadora para que el administrador o persona encargada de la granja, pueda acceder y transferir la información del ganado bovino, desde un cliente FTP como WinSCP hacia el servidor FTP instalado en la minicomputadora.

Revisar en el Manual del Administrador la instalación del servidor FTP (vsftpd) y el uso del cliente FTP (Filezilla).

3.3 REQUERIMIENTOS TECNOLÓGICOS (COMPONENTES ELECTRÓNICOS Y RECURSOS UTILIZADOS)

A continuación se realizará una justificación de los elementos más relevantes del diseño del sistema, se detallan los recursos utilizados de los mismos:

3.3.1 ARDUINO MEGA 2560

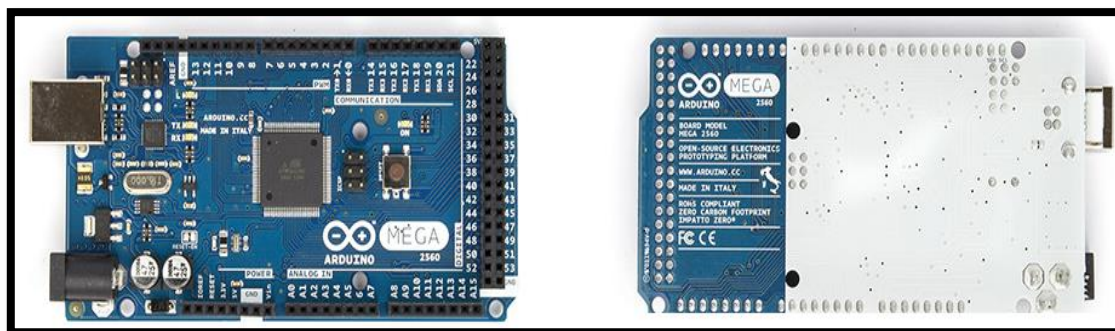


Figura 37 Arduino Mega 2560

Fuente: <http://www.arduino.cc/en/Main/ArduinoBoardMega2560>

De entre todas las opciones disponibles se ha optado por escoger el dispositivo Arduino Mega 2560, como se puede observar este microcontrolador es una de las plataformas más potentes y el que más pines de entrada/salida posee, por lo cual es apto para aplicaciones ya algo complejas. Cuenta con más memoria para el programa, más RAM.

Además satisface con los requerimientos que se necesita para este proyecto de titulación: como es el protocolo de comunicación SPI, 4 protocolos seriales (UART), cuenta

con 54 pines digitales y 16 analógicos. Estos pines satisfacen la necesidad de conectar múltiples componentes.

Existe una amplia información del dispositivo lo que conlleva a una gran comunidad. La placa ofrece muchas prestaciones que lo capacitan para desempeñar la función de microcontrolador para este proyecto como son: consumos de energía reducidos y posee un bajo coste del dispositivo.

3.3.1.1 Recursos utilizados de Arduino Mega 2560

- Comunicación serial (UART) para comunicar con la minicomputadora.
- Comunicación SPI para comunicar con el módulo RFID.
- Entradas digitales para el pulsador.
- Salidas digitales para leds, luces pilotos, motor eléctrico y la válvula solenoide.
- Funciones especiales interrupciones para el sensor de flujo.

3.3.2 Raspberry Pi B+.

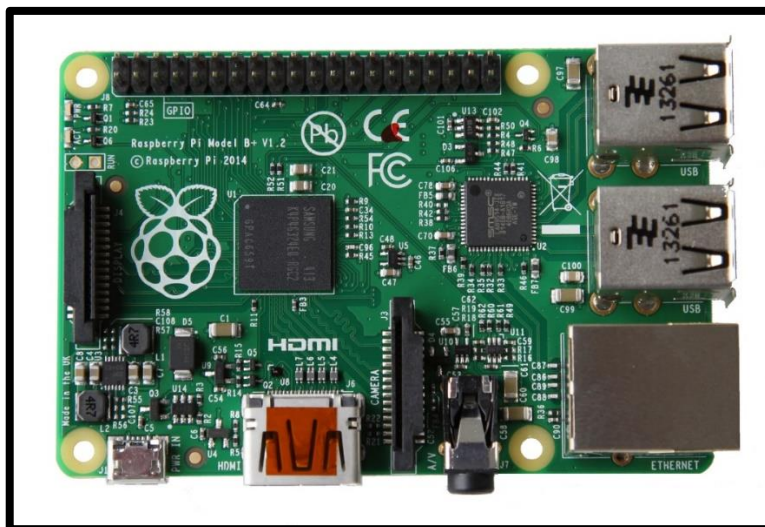


Figura 38 Raspberry Pi B+
Fuente: <https://www.raspberrypi.org/>

La minicomputadora seleccionada para el proyecto fue Raspberry Pi B+ ya que además de poseer unas características fundamentales como: tamaño reducido, bajo coste y consumo de energía reducidos. Esta minicomputadora posee el beneficio de que su sistema operativo está basado en Debian, por ello se lo denomina Raspbian.

3.3.2.1 Recursos utilizados de la mini computadora Raspberry Pi B+.

- Comunicación Serial (UART) para comunicar con el microcontrolador.
- Comunicación I2C para comunicar con el módulo RTC.
- Entrada Digital para el pulsador.
- Servidor LAMP. (Linux, Apache, MySQL, PHP, Python).
- phpMyAdmin para administrar la base de datos MySQL a través de una interfaz web.
- Servidor FTP.
- Servidor SSH.

- Servidor VNC.

3.3.3 MODULO RFID RC522

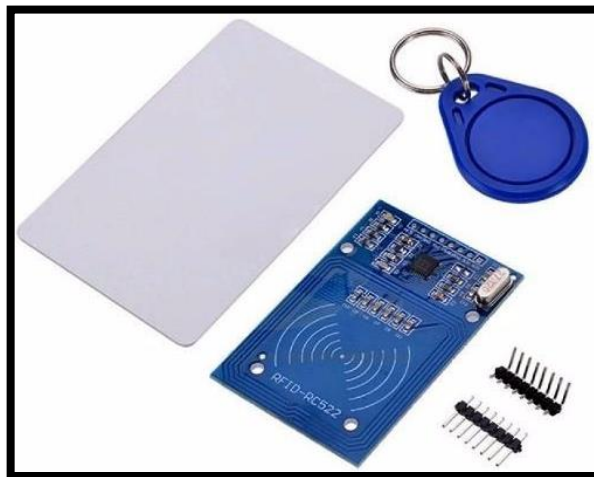


Figura 39 Módulo RFID RC522

Fuente:

http://fotos.pccomponentes.com/electronica/arduino/modulo_rfid_rc522_con_tarjeta_ic_s50_compatible_con_arduino.jpg

El módulo utiliza 3.3 V como voltaje de alimentación y se controla a través del protocolo de comunicación SPI, por lo que es compatible con cualquier microcontrolador. Este módulo utiliza un sistema avanzado de modulación y demodulación para todo tipo de dispositivos pasivos de 13.56 Mhz. Este módulo es el encargado de la identificación de cada bovino.

3.3.3.1 Características del módulo RFID RC522

- Modelo: MF522-ED
- Corriente de operación: 13-26mA a 3.3V
- Corriente de standby: 10-13mA a 3.3V
- Corriente de sleep-mode: <80 μ A

- Corriente máxima: 30mA
- Frecuencia de operación: 13.56Mhz
- Distancia de lectura: 0 a 60mm
- Protocolo de comunicación: SPI
- Velocidad de datos máxima: 10Mbit/s
- Dimensiones: 40 x 60 mm
- Temperatura de operación: -20 a 80°C
- Humedad de operación: 5%-95%
- Máxima velocidad de SPI: 10Mbit/s

3.3.4 SENSOR DE FLUJO FS300A



Figura 40 Sensor de flujo FS300A

Fuente: http://www.seeedstudio.com/depot/g34-water-flow-sensor-p-1083.html?cPath=144_151

Este sensor consiste en un cuerpo de plástico como válvula, un rotor de líquido, y un sensor de efecto Hall. Cuando el líquido fluye a través del rotor, el rotor rueda. El sensor brinda una frecuencia proporcional al flujo de líquido. El sensor de efecto Hall emite la señal de impulso correspondiente. Este sensor tiene la función de medir la cantidad de flujo que fluye por el mismo.

3.3.4.1 Características del sensor

- Compacto y fácil de instalar.
- Sensor de efecto Hall de alta calidad.
- Cuenta con 3 pines: Rojo (Voltaje positivo), Negro (Tierra) y Amarillo (salida de frecuencia.)

3.3.4.2 Especificaciones del sensor

- Min. Voltaje de funcionamiento: 4.5V DC
- Max. Corriente de operación: 15 mA (5V DC)
- Voltaje de funcionamiento: 5V ~ 24V DC
- Caudal Q: 1 ~ 60 [litros/min]
- Capacidad de carga: ≤ 10 mA (DC 5V)
- Temperatura de funcionamiento: ≤ 80 °C
- Temperatura del líquido: ≤ 120 °C
- Humedad de funcionamiento: 35% ~ 90% RH
- Presión del agua: ≤ 1.75 MPa
- Temperatura de operación: -25 ~ + 80 °C
- Humedad de almacenamiento: 25% ~ 95% RH
- Exactitud: el 10%

3.3.5 VÁLVULA SOLENOIDE



Figura 41 Válvula Solenoide

Fuente: http://www.olimex.cl/product_info.php?products_id=726

Este tipo de válvulas solenoides eléctricas son usadas para proyectos que implican el control de fluido de baja viscosidad.

3.3.5.1 Características de la Válvula solenoide

- Normalmente cerrado.
- Durable.
- Filtro de entrada de Acero inoxidable.
- Temperatura de funcionamiento amplia.

3.3.5.2 Especificaciones de la Válvula solenoide

- Voltaje de funcionamiento: 12V DC
- Corriente máxima: 450mA
- Modo de funcionamiento: normalmente cerrado

- Puertos de entrada/salida: G1/2"
- Materiales: nylon/acero inoxidable/Polioximetileno
- Tipo de Vale: válvula de diafragma (Operado por servo)
- Pantalla de filtro: filtro de entrada de acero inoxidable
- Temperatura máxima de funcionamiento: 120 ° C
- Rango de Presión de funcionamiento: 0.02-0.8 Mpa
- Vida prevista: 200,000 +
- Uso: agua y fluidos de baja viscosidad

3.3.6 PANTALLA GRAFICA GLCD 128X64

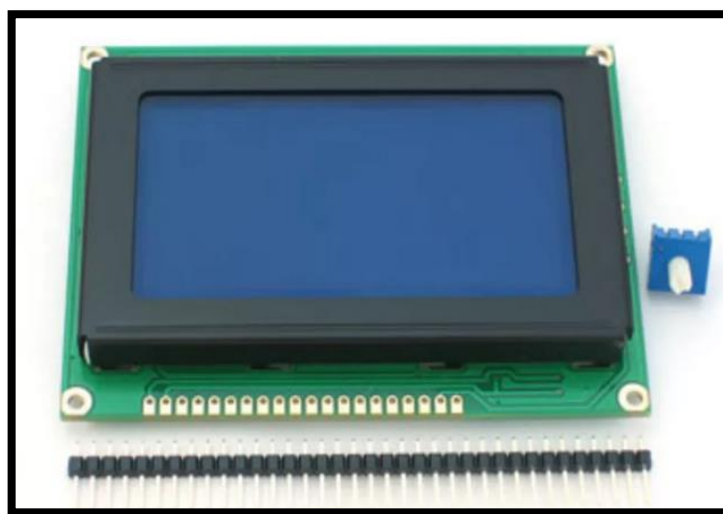


Figura 42 Parte frontal de pantalla grafica GLCD 128x64

Fuente: http://static3.tme.eu/products_pics/9/8/c/98c1c00cd04439ed3b45ee3d3060e27a/293295.jpg

Los LCD's gráficos (GLCD por sus siglas en inglés) son pantallas de cristal líquido cuya resolución es medida en pixeles. Existen en el mercado electrónico una gran diversidad de display's gráficos, de tamaños y tipos variados.

Un GLCD por sí mismo no se puede conectar directamente a un microcontrolador, para ello necesita de un chipdriver, que se encarga de interpretar las señales recibidas. Los chipdrivers más empleados son: KS0108 (A, B, y C), SED133x, T6963c, Noritake3900 y lc7891. Cada chip requiere de un circuito específico que actúa como interfaz entre el microcontrolador y el LCD.

El GLCD tiene pines de control y de datos. Los de control son: CS1, CS2, DI, RW, E y Reset. Los de datos son del DB0 al DB7. Este GLCD se divide en 2 zonas de 64 * 64 pixeles. Con los pines de selección CS1 y CS2 seleccionamos que zona queremos usar.

Con los pines DI y RW nos sirven para controlar el display y decirle si queremos leer o escribir datos desde o hacia el display, ejecutar alguna instrucción o leer el status del GLCD.

El pin E (enable) sirve como forma de decirle al display que lea los otros pines de control y ejecute lo que se le indica por medio de estos. Con los pines de datos se va poder enviar o recibir datos del display y leer el status del chip de control del display.

3.3.6.1 Características de una GLCD 128x64

- Conformado por una matriz de puntos de visualización de 128 pixeles de largo por 64 pixeles de alto.
- Su iluminación de fondo está entre verde-amarillo cuando se enciende.
- Fácil manejo.

- Bajo consumo.
- Contiene dos controladores internos un KS0108B y KS0107B.

3.3.6.2 Descripción de los pines de conexión de una GLCD 128x64

PIN 1: VSS (Conexión a tierra)

PIN 2: VDD (Conexión de alimentación ± 5 V)

PIN 3: V0 (Voltaje de salida negativa, por lo general es usada con un potenciómetro con el PIN 18 para el ajuste del contraste de los pixeles)

PIN 4: D/I (Datos de E/S para el cambio de registro)

SPIN 5: R/W (Determina si los datos se van a leer o escribir)

PIN 6: E (Habilita la comunicación con la GLCD)

PIN 7 - 14 (Especifica un dato de 8-Bits de información)

PIN15: CS1 (Indica si se selecciona la primera mitad de la pantalla, pixeles 0-63)

PIN16: CS2 (Indica si se selecciona la segunda mitad de la pantalla, pixeles 64-127)

PIN17: RESETB (Señal de reinicio, funciona de varias forma dependiendo de la ocasión)

PIN18: VEE (Conexión de ajuste de contraste de los pixeles)

PIN19: A (Conexión positiva de la luz de fondo, por lo general son +5V)

PIN20: K (Conexión negativa de la luz de fondo, por lo general es tierra)

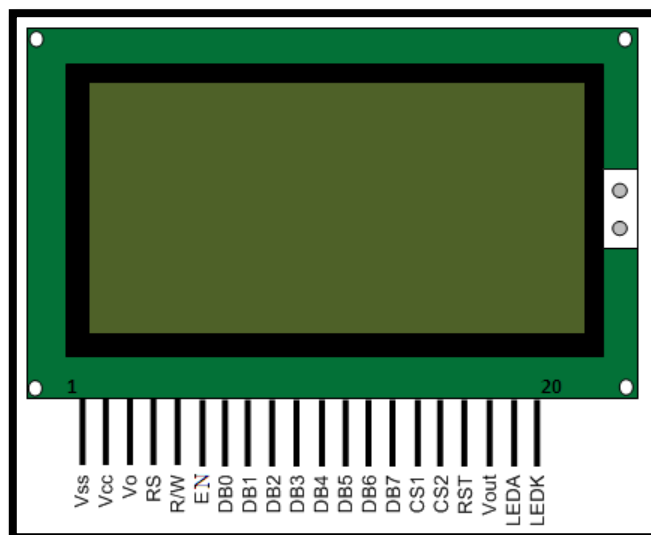


Figura 43 Pines de la GLCD 128X64

Fuente: <http://www.engineersgarage.com/electronic-components/graphics-lcd>

3.3.7 MOTOR ELÉCTRICO

El motor eléctrico utilizado para el retirado automático es el motor para plumas de un camión Hyundai HD72 de 24V DC. Este motor tiene el suficiente torque para el retiro de las pezoneras, además posee un sistema reductor de velocidad para retirar con seguridad el juego de ordeño.



Figura 44 Motor de plumas de camión Hyundai HD72

Fuente: <http://www.engineersgarage.com/electronic-components/graphics-lcd>

3.3.8 MODULO RTC (REAL TIME CLOCK) 3231

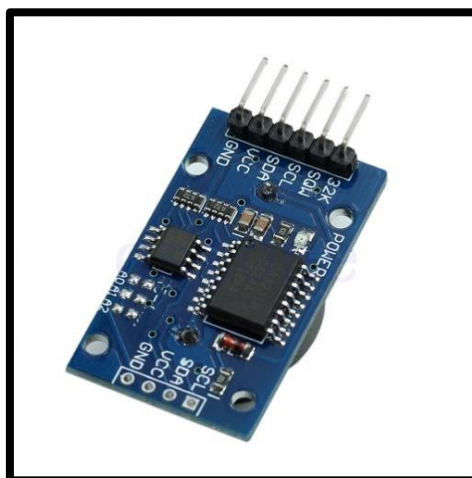


Figura 45 Modulo RTC 3231

Fuente: <http://www.electrotekmega.com>

La Raspberry Pi B+ mientras esté conectada a la red con Internet establece una conexión con el servidor NTP (Network Time Protocol) para ajustar la hora del sistema. Pero en caso de no estar conectado y para evitar fallas en el tiempo y fecha se optó por instalar un módulo RTC (Real Time Clock). El módulo RTC resolverá el problema, así cuando la Raspberry Pi B+ se inicie utiliza la fecha y hora de este módulo para establecer correctamente dichos parámetros en su sistema.

Este módulo utiliza la comunicación I2C (Circuitos Inter-Integrados), y es utilizado para conectar periféricos de bajas velocidades.

3.3.9 SERVIDOR LAMP

Es un paquete con todo lo necesario para poder instalar un servidor web con la mayoría de las funcionalidades que vamos a necesitar.

3.4 DISEÑO DEL HARDWARE DEL SISTEMA

3.4.1 Arquitectura del Sistema

El presente esquema representa de manera general el proyecto a desarrollarse.

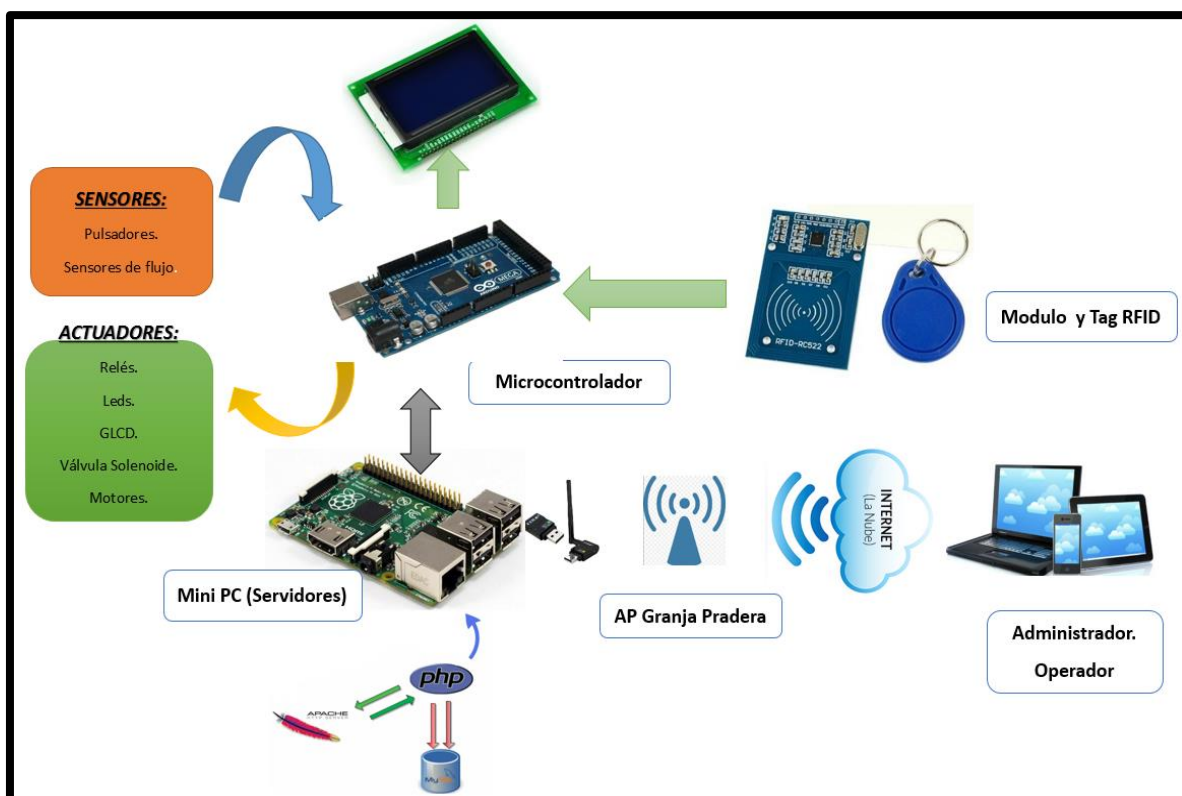


Figura 46 Arquitectura del sistema
Fuente: Autor

3.4.2 Diagramas de bloques

En el siguiente diagrama se representa la estructura y el modo en que se relacionan los subsistemas que se han tomado en cuenta para conformar un sistema funcional.

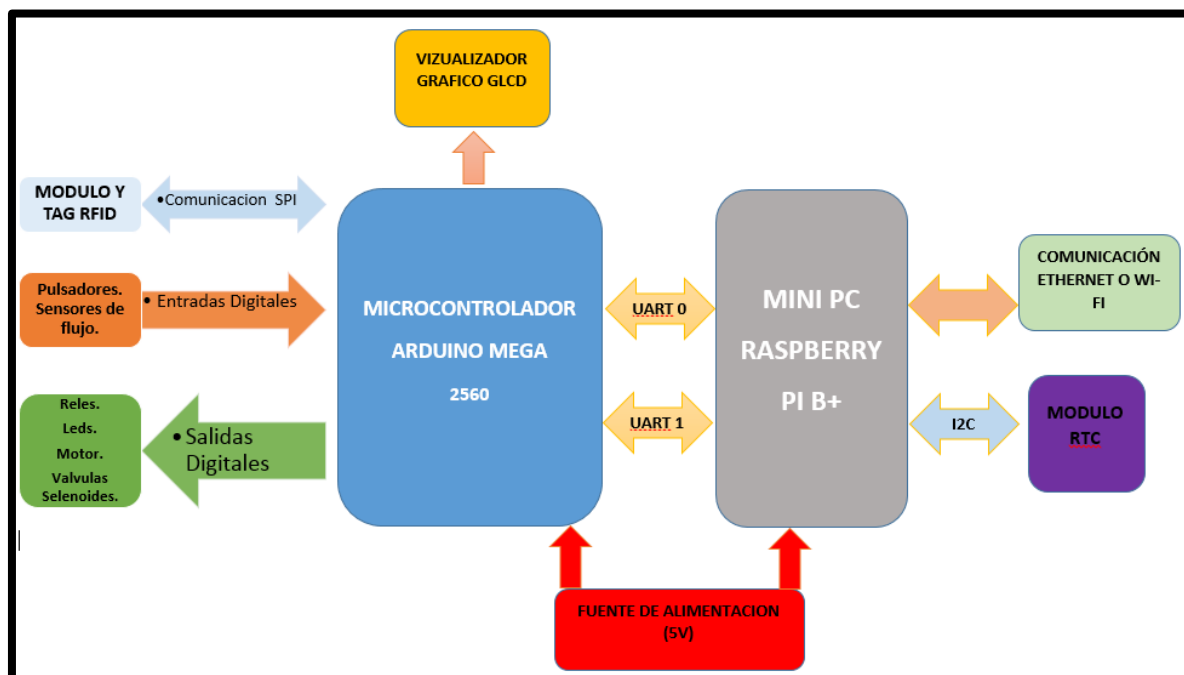


Figura 47 Diagramas de bloques del sistema
Fuente: Autor

3.4.3 Diagrama de conexión (esquemático)

Para que los diagramas de conexión se puedan esquematizar de una manera didáctica, se ha optado por elegir el software Fritzing, el cual me permite documentar los prototipos para después crear los esquemas de circuitos impresos.

Se presenta el diagrama de conexión para realizar las pruebas con el microcontrolador Arduino Mega 2560 y la pantalla grafica GLCD 128x64.

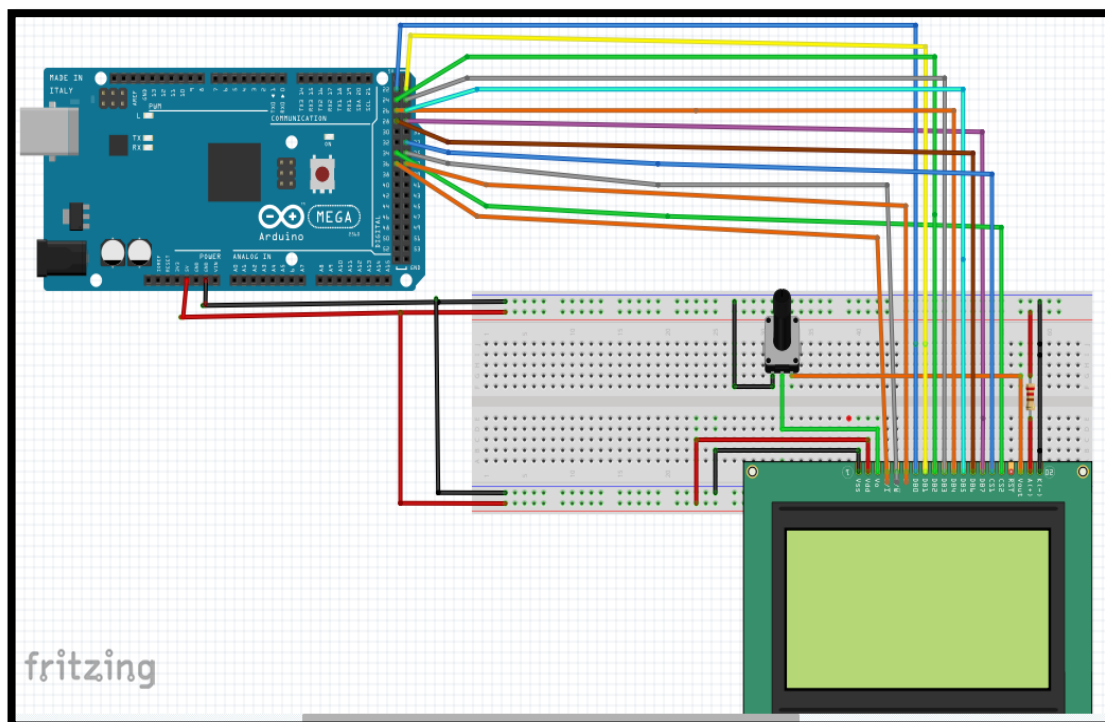


Figura 48 Diagrama de conexión entre Arduino Mega 2560 y pantalla grafica GLCD 128x64
Fuente: Autor

Como se puede observar en la figura 49 se presenta el diagrama de conexión para realizar las pruebas con el microcontrolador Arduino Mega y el módulo RFID (RC522). La comunicación entre estos dos dispositivos se lo realiza a través del bus SPI (Serial Peripheral Interface).

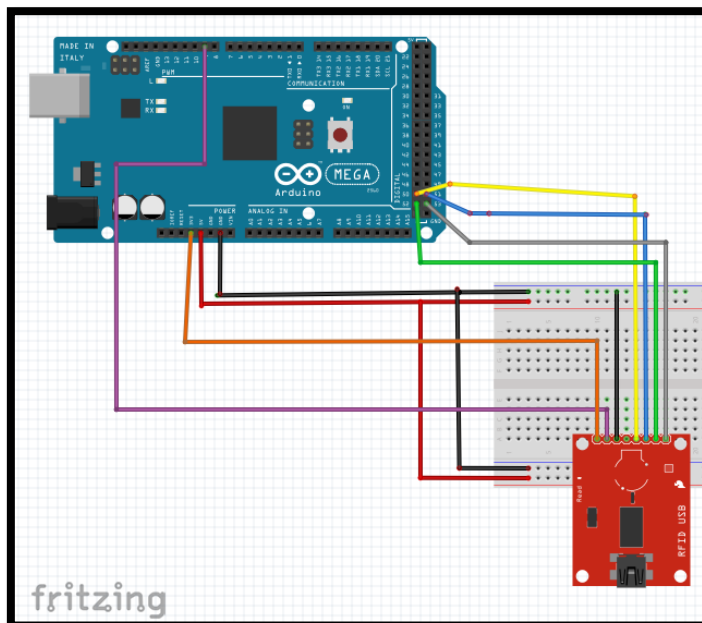


Figura 49 Diagrama de conexión entre Arduino Mega 2560 y lector RFID RC522
Fuente: Autor

A continuación se muestra el diagrama de conexión para realizar las pruebas con el microcontrolador Arduino Mega y la minicomputadora Raspberry Pi B+. La comunicación entre estos dispositivos de propósito general se lo realiza a través del bus UART (Universal Asynchronous Receiver/Transmitter).

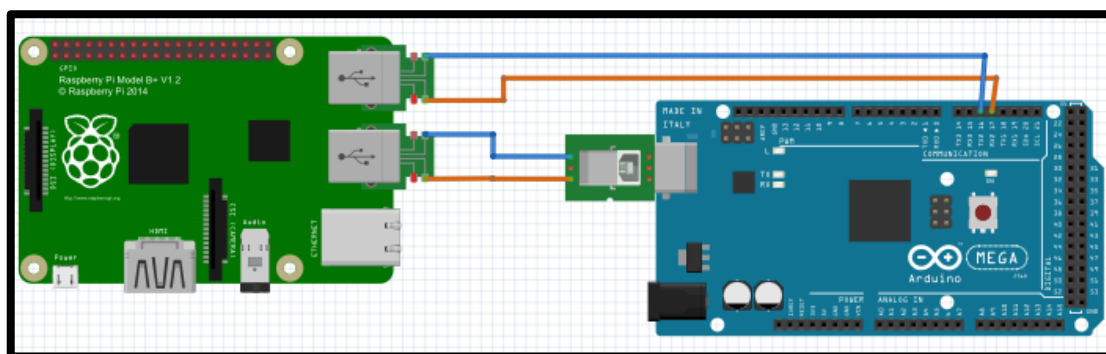


Figura 50 Diagrama de conexión entre Arduino Mega 2560 y Raspberry Pi B+
Fuente: Autor

A continuación se detalla el diagrama de conexión, para realizar las pruebas con el microcontrolador Arduino Mega y los dispositivos de activación del sistema, como el pulsador, los dispositivos de visualización del sistema tanto internos (led) como externos

(luz pilotos). Se detalla también la parte de electrónica de potencia basada en relés, para la utilización de la válvula solenoide y el motor eléctrico para el retiro automático de las pezoneras del animal.

Cabe mencionar que para el control del giro del motor eléctrico, se realizó un puente H basado en relés, los cuales son activados por sus respectivos transistores. La señal de activación de los transistores lo realizan los pines del microcontrolador Arduino Mega 2560.

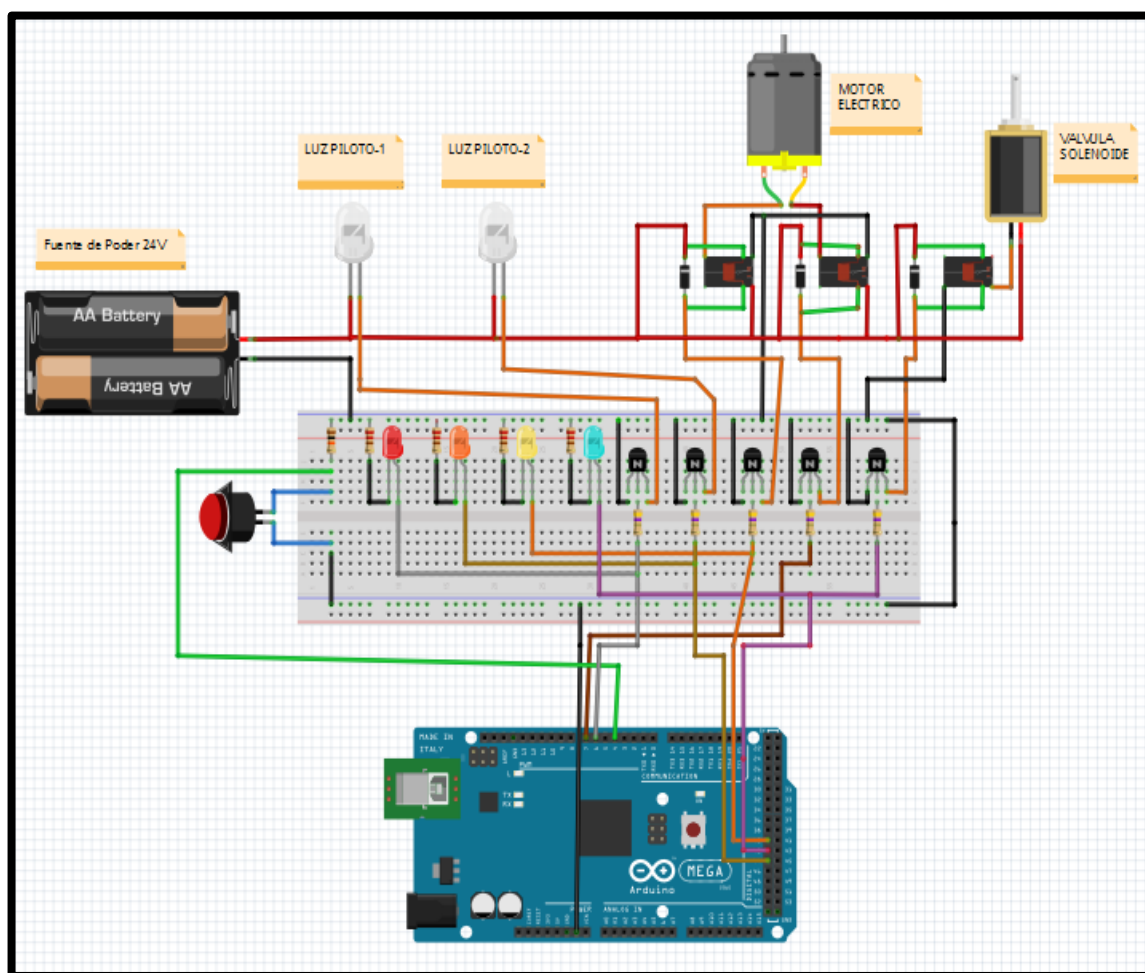


Figura 51 Diagrama de conexión del Arduino Mega 2560 y los demás sistemas
Fuente: Autor

En la figura 52 se puede observar el diagrama de conexión entre el módulo RTC y la minicomputadora Raspberry Pi B+.

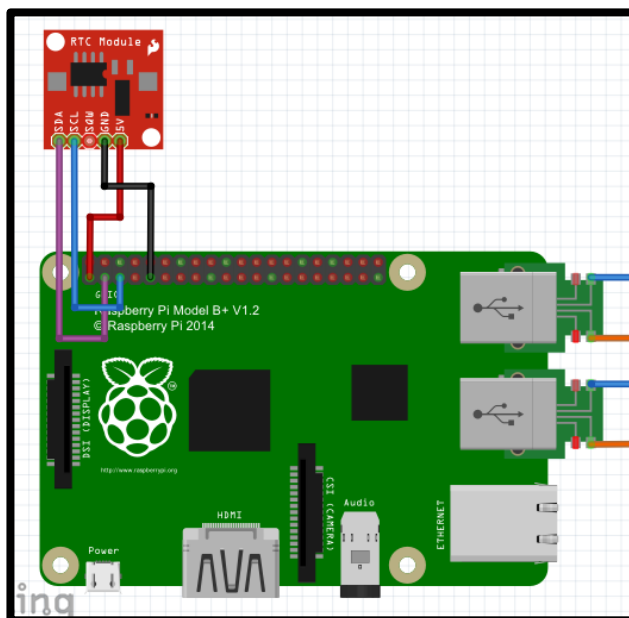


Figura 52 Diagrama de conexión del módulo RTC y Raspberry Pi B+
Fuente: Autor

Para que el sistema tenga un apagado automático, se hace uso del puerto GPIO (General Purpose Input Output) de la Raspberry PI B+, este puerto GPIO es utilizado como entrada para el pulsador. Una vez que es activado el pulsador, se ejecuta un comando que apaga el sistema. El comando que se utilizó es `shutdown -h now`, en el manual del administrador se puede observar el script realizado en lenguaje Python, y su respectiva automatización. El diagrama de conexión del pulsador para apagado del sistema es el siguiente:

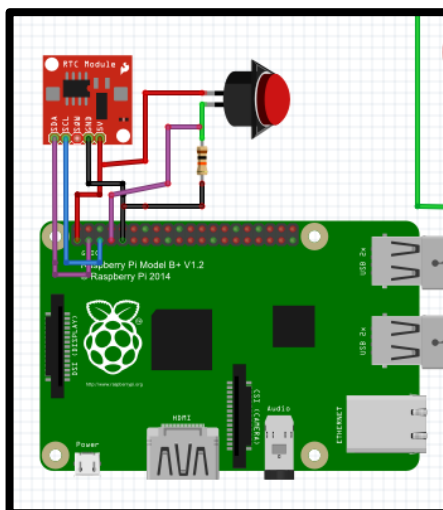


Figura 53 Diagrama de conexión del pulsador y Raspberry Pi B+
Fuente: Autor

Por último se presenta el diagrama de conexión de todos los dispositivos que intervienen en el sistema completo.

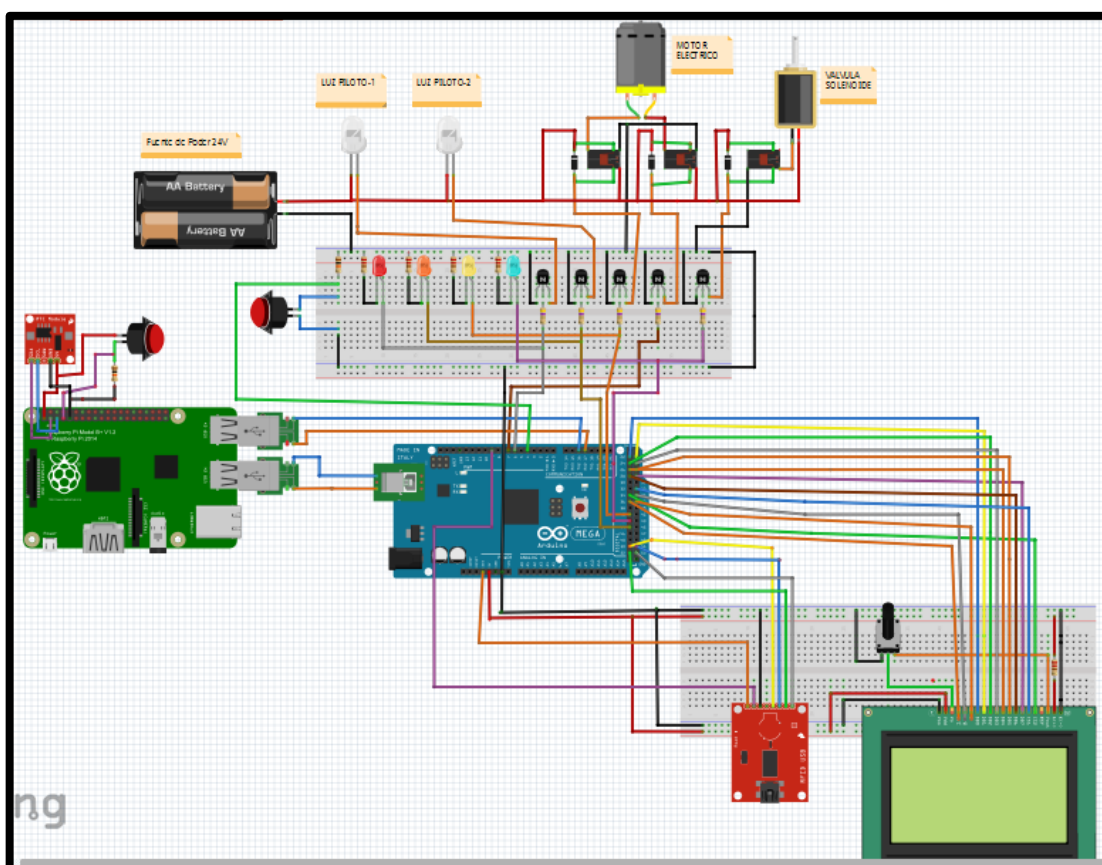


Figura 54 Diagrama de conexión del Sistema completo
Fuente: Autor

3.4.4 Distribución de pines del módulo Arduino Mega 2560

Para especificar la utilización de los pines del módulo Arduino mega se detalla una tabla con el número de pin, si el pin utilizado es entrada se utilizará la notación de I, si es salida se utilizará la notación de O, si el tipo de dato que se utilizó es Digital o Analógico, se detalla la función y la especificación que cumple determinado pin del Arduino con los demás componentes electrónicos.

Tabla 15 *Distribución de pines del microcontrolador Arduino Mega 2560 con los demás dispositivos*

PIN ARDUINO	I/O	TIPO DE DATO	ESPECIFICACIÓN/FUNCIÓN
22	O	Digital	DB0 GLCD
23	O	Digital	DB1 GLCD
24	O	Digital	DB2 GLCD
25	O	Digital	DB3 GLCD
26	O	Digital	DB4 GLCD
27	O	Digital	DB5 GLCD
28	O	Digital	DB6 GLCD
29	O	Digital	DB7 GLCD
33	O	Digital	CS1 GLCD
34	O	Digital	CS2 GLCD
35	O	Digital	R/W GLCD
36	O	Digital	D/I GLCD
37	O	Digital	E GLCD
50	I	Digital	MISO RFID RC-522
51	O	Digital	MOSI RFID RC-522
52	O	Digital	SCK RFID RC-522
53	O	Digital	SDA RFID RC-522
9	O	Digital	RST RFID RC-522
0	I	Digital	RX0 Raspberry Pi B+
1	O	Digital	TX0 Raspberry Pi B+
4	I	Digital	PULSADOR 1
6	O	Digital	LED ON/OFF SISTEMA
2	I	Digital	SENSOR DE FLUJO 1 (INTERRUPCIONES)
40	O	Digital	ACTUADOR VÁLVULA SOLENOIDE 1
42	O	Digital	RELÉ MOTOR 1
7	O	Digital	RELÉ MOTOR 2

44

O

Digital

RELÉ SEÑAL SONORA O VISUAL

Fuente: Autor**Notación:**

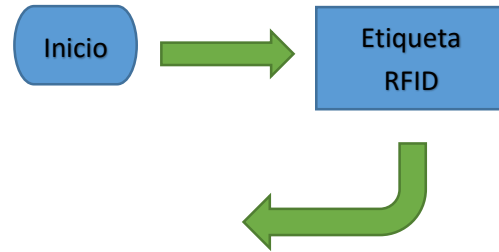
I ENTRADA

O SALIDA

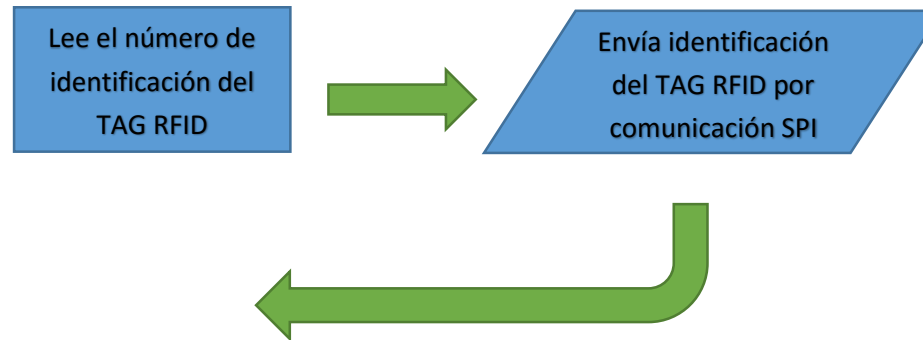
3.5 DISEÑO DEL SOFTWARE DEL SISTEMA

3.5.1 Diagramas de flujo

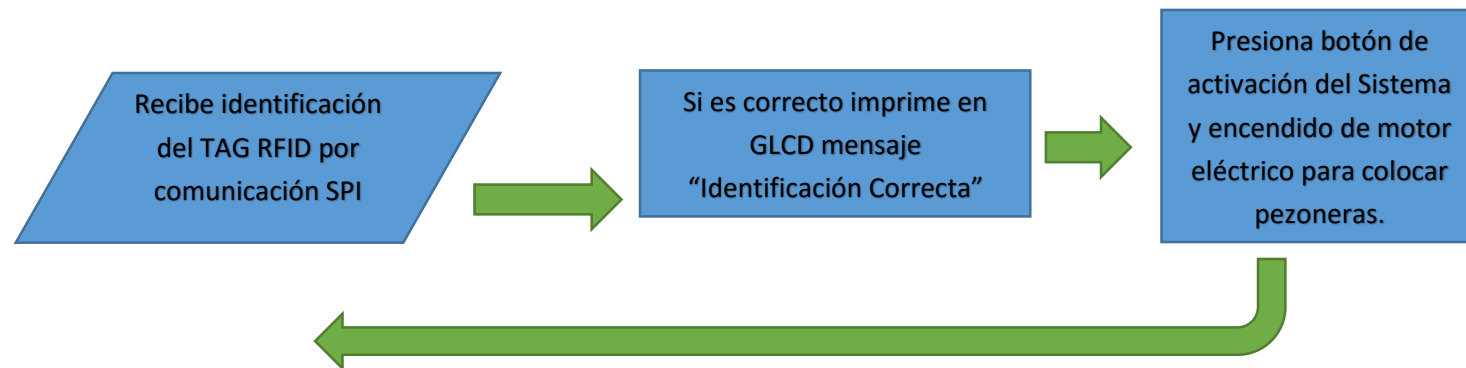
TAG RFID

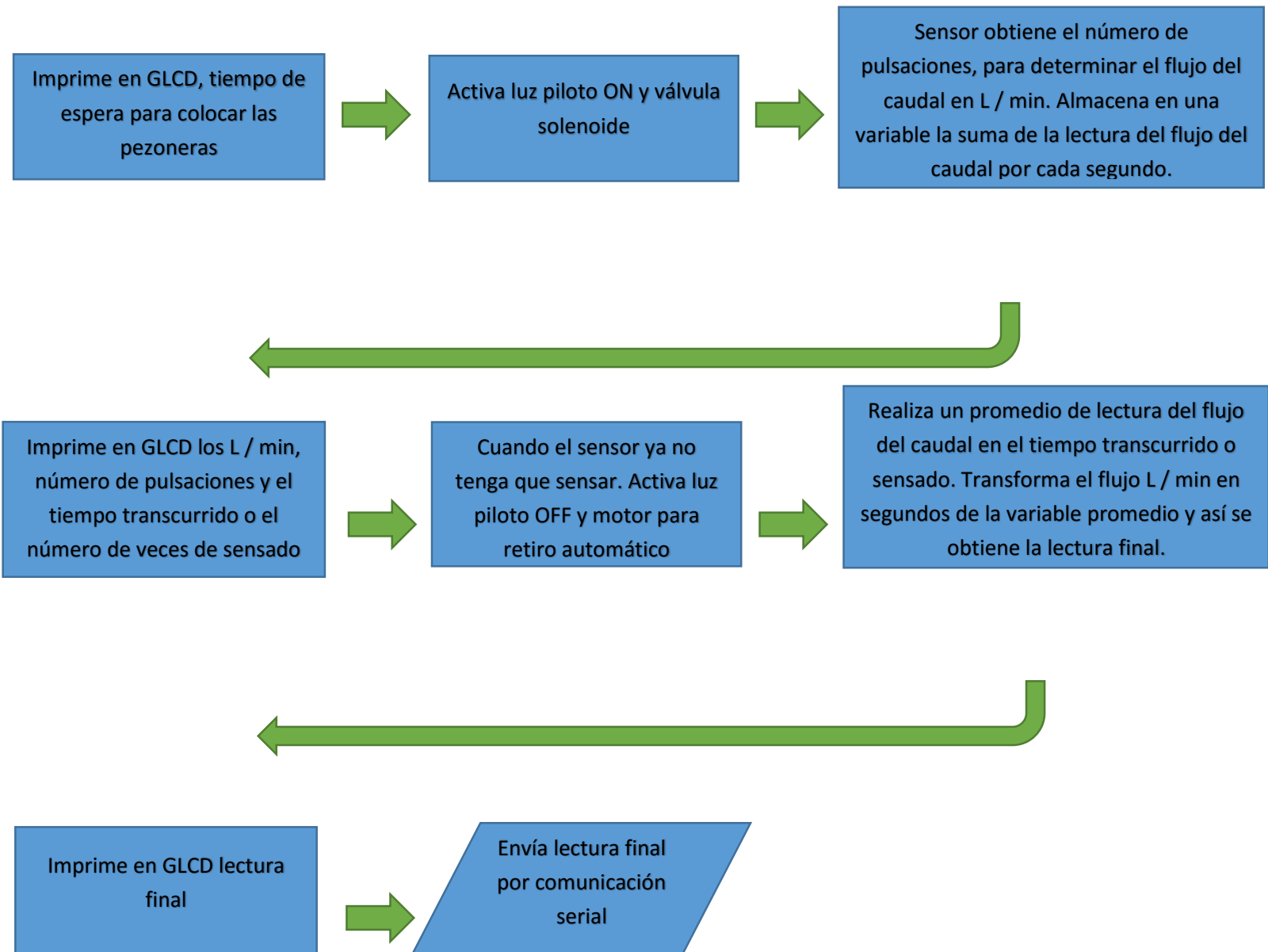


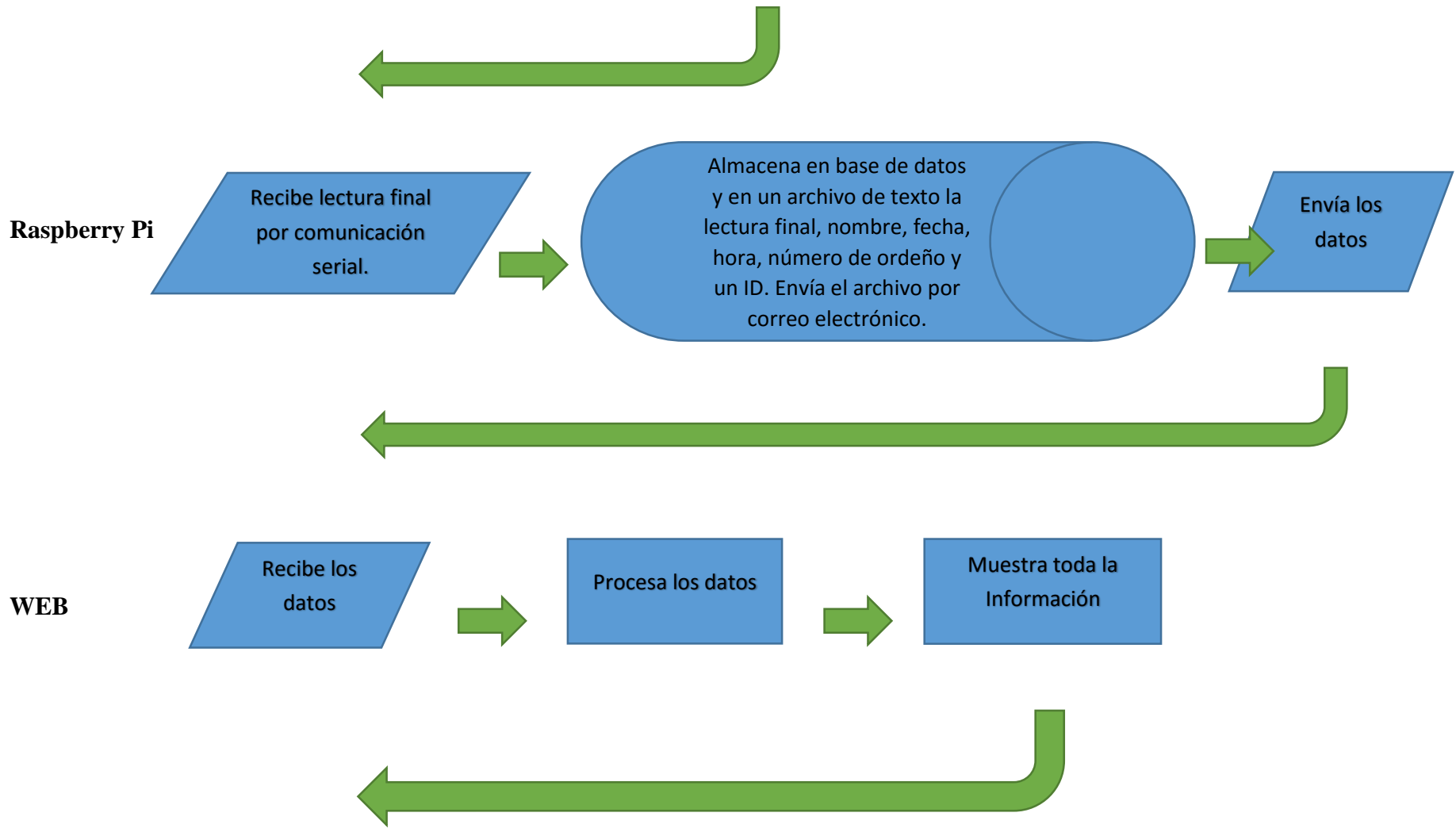
LECTOR RFID



ARDUINO MEGA







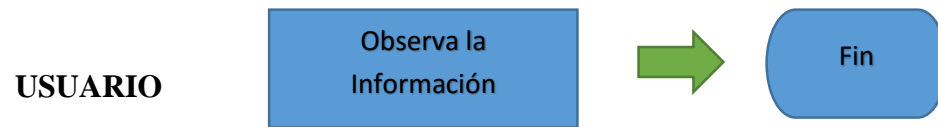


Figura 55 Diagrama de flujo del sistema
Fuente: Autor

Desde un enfoque general podemos decir que el sistema que hemos propuesto para cumplir con los objetivos, se soluciona con este diagrama de bloques. El módulo Arduino tiene el rol de identificar cada etiqueta RFID, y de controlar las entradas y salidas necesarias para el funcionamiento del sistema, además tiene la función de enviar a la minicomputadora el resultado obtenido del proceso de extracción de leche de cada bovino. La minicomputadora tiene el rol de almacenar en una base de datos y en un archivo de texto, el resultado obtenido del módulo Arduino y otros datos importantes. Además en la minicomputadora se procederá habilitar un servidor Web, para gestionar la base de datos obtenida, así como consultar resultados previos del ganado bovino a través de una interfaz gráfica. En la minicomputadora además se instalará un servidor FTP el cual permite la transferencia de archivos.

3.5.2 Algoritmo de programación

Debido a lo extenso que es el firmware se ha tomado la decisión de colocarlo en los Anexos.

En el firmware del microcontrolador Arduino se utilizó las siguientes librerías para facilitar el desarrollo del algoritmo de programación del sistema:

- `#include <SPI.h>`: esta librería es la encargada de habilitar la comunicación SPI entre el microcontrolador Arduino y el módulo lector RFID.
 - En el setup del código de programación del Arduino hay que iniciar y configurar el SPI, con el comando `SPI.begin()`.
- `#include <RFID.h>`: esta librería es la encargada de inicializar el proceso de lectura de las etiquetas RFID.
 - Además en el setup del código de programación del Arduino para que se inicialice la librería RFID hay que escribir el comando `rfid.init()`.
- `#include <glcd.h>`: con esta librería se puede hacer uso para la escritura en la pantalla grafica GLCD de 128x64.
 - Para que se inicialice la librería en el setup del código de programación del Arduino se debe escribir `GLCD.Init()`.
- `#include <fonts/allFonts.h>`: es la librería encargada de seleccionar el tipo de fuentes para la escritura en la pantalla grafica GLCD 128x64.

En el firmware de la minicomputadora se importó los siguientes módulos o librerías en un script del lenguaje de programación de Python, para facilitar el desarrollo del algoritmo de programación del sistema:

- `import MySQLdb as mdb`: me permite conectar a un servidor de base de datos MySQL desde Python.
- `import time`: este módulo ofrece varias funciones relacionadas con el tiempo.
- `import datetime`: este módulo provee clases para manipular fechas y horas.
- `import serial`: módulo que me permite habilitar un puerto serial.
- `import os`: este módulo permite ejecutar comandos externos pertenecientes a la Raspberry, para el pagado correcto de todo el sistema, y él envió del archivo de texto por correo electrónico.

3.6 DISEÑO DE PLACAS DE CIRCUITOS IMPRESOS

Para el diseño de las placas de circuitos impresos, se optó por utilizar el software Eagle (Easily Applicable Graphical Layout Editor) versión 6.4.0, debido a que se puede diseñar diagramas y los respectivos PCBs. Además del orden y mejora de eficiencia que se puede ofrecer en el routeo con prioridades.

3.6.1 Esquemático de conexión de la placa principal

El siguiente esquema de conexión muestra el diseño de las entradas y salidas digitales, leds, fuentes de alimentación, resistencias, diodos de protección, transistores y relés.

Además se ha considerado por colocar borneras para el sensor de flujo, la válvula solenoide, el motor eléctrico, las luces pilotos y las fuentes de alimentación.

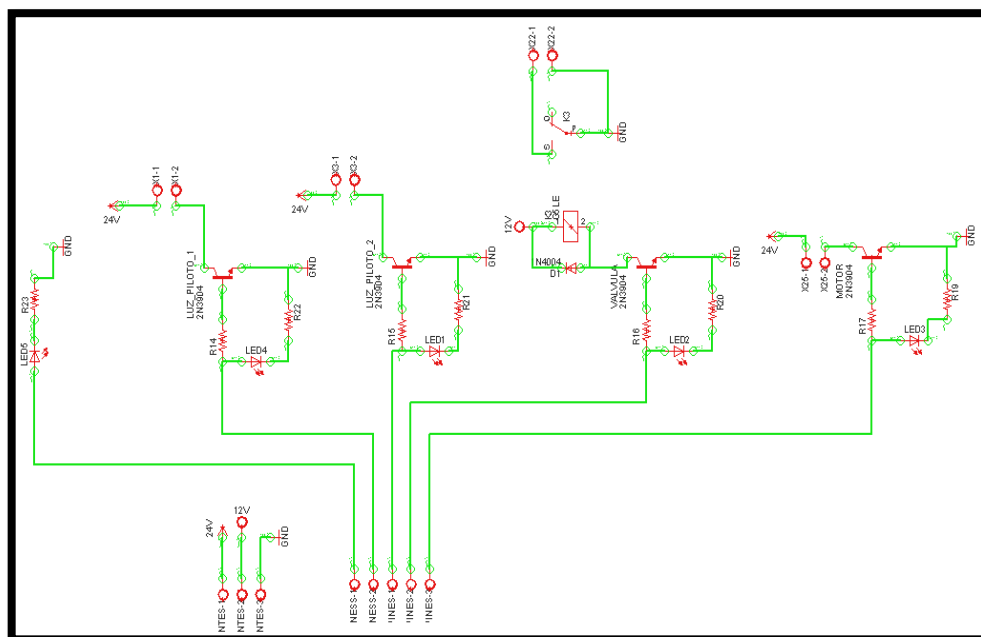


Figura 56 Esquema de conexión del sistema

Fuente: Autor

3.6.2 Placa principal (componentes electrónicos)

Cabe mencionar que el diseño de las placas se lo realizó de manera manual. En la figura 57 se puede observar el diseño del circuito impreso de los componentes electrónicos del sistema.

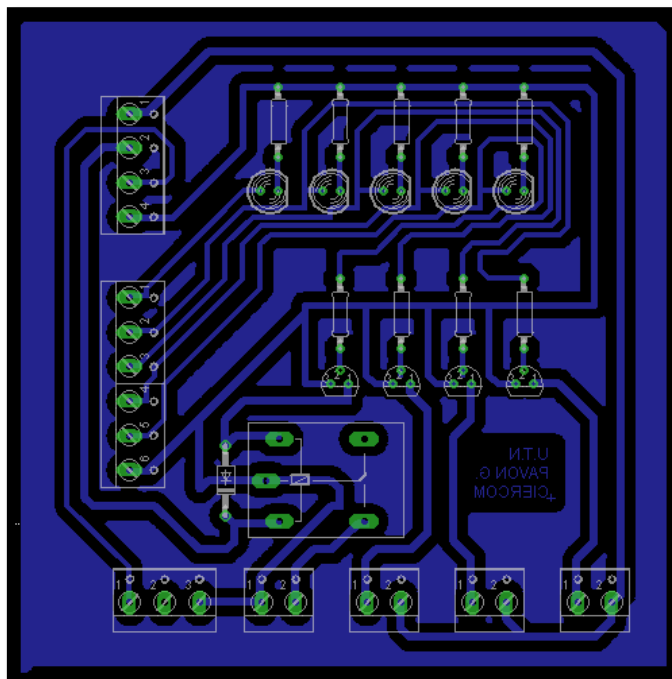


Figura 57 Diseño de la placa principal
Fuente: Autor

En la siguiente tabla 16 se describe los componentes eléctricos utilizados y su voltaje de operación.

Tabla 16 *Voltaje de funcionamiento de los componentes electrónicos utilizados*

Componente Electrónico	Voltaje de operación
Sensor de flujo.	5V
Válvula Solenoide	12V
Motor Eléctrico	24V
Luces Pilotos.	24V

Fuente: Autor

Como la válvula solenoide tiene un voltaje de operación de 12 V. Se diseñó una placa reguladora de voltaje de 12V con el integrado 7812 para alimentar a la válvula.

3.6.3 Placa secundaria

El diseño de esta placa consta de la fuente de alimentación de 12 V y del botón de activación del sistema el cual es alimentado con 5V provenientes del módulo Arduino.

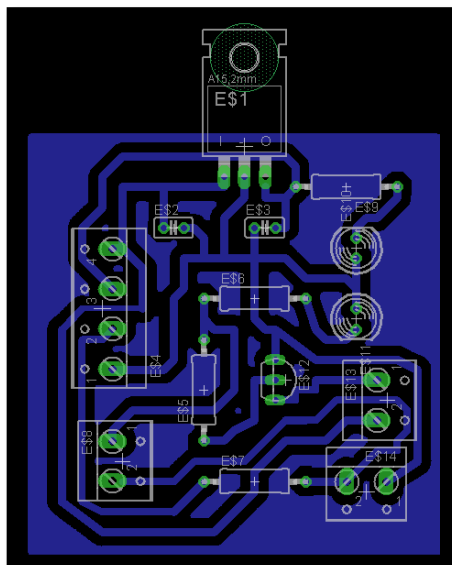


Figura 58 Diseño de la placa secundaria
Fuente: Autor

3.7 Montaje final en Protoboard

Como se puede observar en la figura 59 se muestra el desarrollo del sistema en la protoboard para realizar las respectivas pruebas.

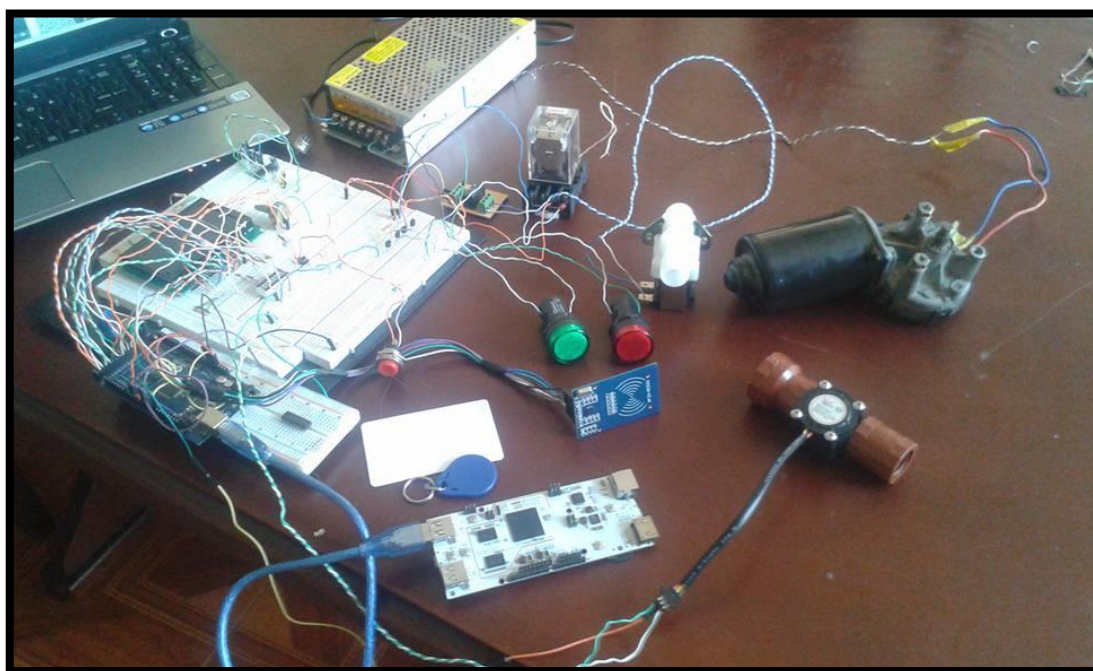


Figura 59 Pruebas previas del sistema en tablero de ensayo electrónico
Fuente: Autor

CAPÍTULO IV

Una vez realizado los diseños de hardware y software del sistema, en este capítulo se procede a la instalación del sistema para realizar las respectivas pruebas, la depuración y corrección de errores tanto de software como de hardware.

4.1 IMPLEMENTACIÓN DEL SISTEMA

4.1.1 Montaje del Gabinete

El montaje de los dispositivos electrónicos se lo realizó en un gabinete de 30 x 30 x 10 cm para darle una estructura modular. Además de proteger y operar directamente los circuitos. Este tablero esta empotrado en la pared bajo techo para prevenir lluvias, humedad, manipulación inadecuada o cualquier otro factor que puedan afectar al sistema.

El gabinete está compuesto también con un sistema de protección para todo el sistema como fusibles y breaker, además de los relés de 12V para activar la válvula solenoide y los relés de 24V para el control del giro del motor eléctrico para el retirado automático de las pezoneras.

En la figura 60 se detallan los elementos que se encuentran en la parte interna del Gabinete.

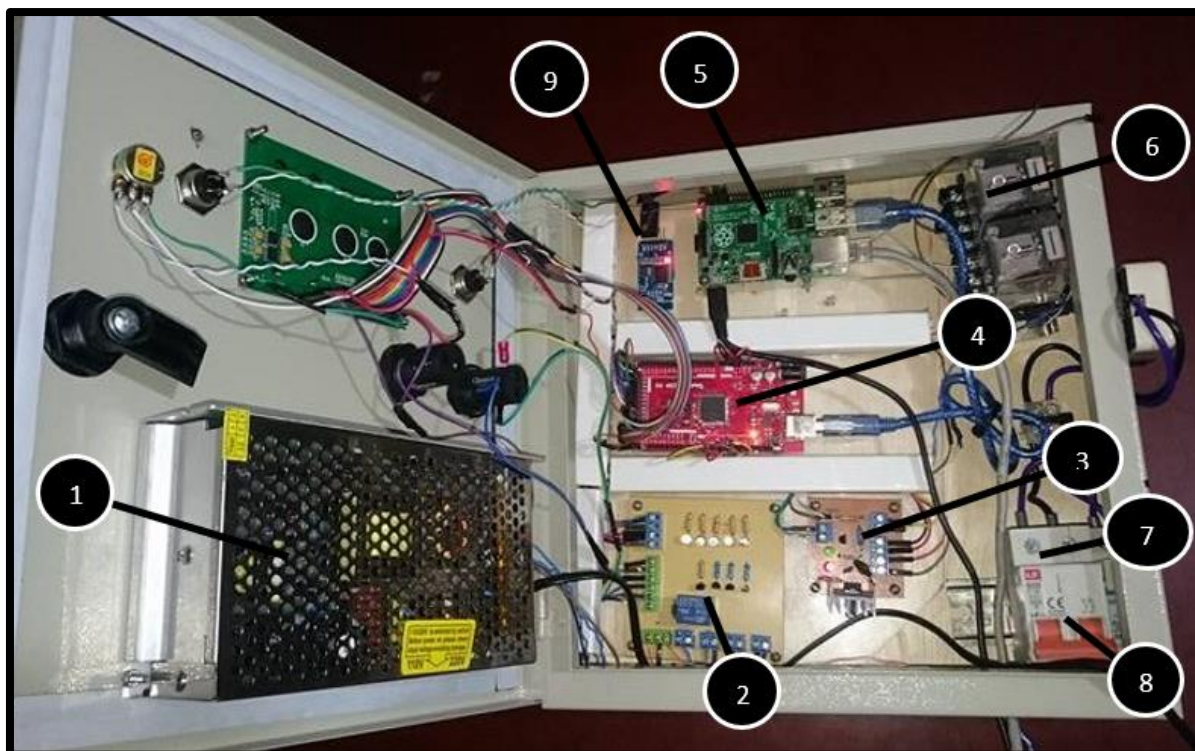


Figura 60 Parte interna del gabinete eléctrico
Fuente: Autor

1. Fuente de alimentación de 24V DC y 6.5A
2. Placa principal.
3. Placa secundaria.
4. Microcontrolador Arduino Mega 2560.
5. Minicomputadora Raspberry Pi B+.
6. Relés electromecánicos de 24V.
7. Breaker.
8. Portafusibles.
9. Módulo RTC.

A continuación se detalla en la figura 61 los elementos de la parte externa del gabinete.

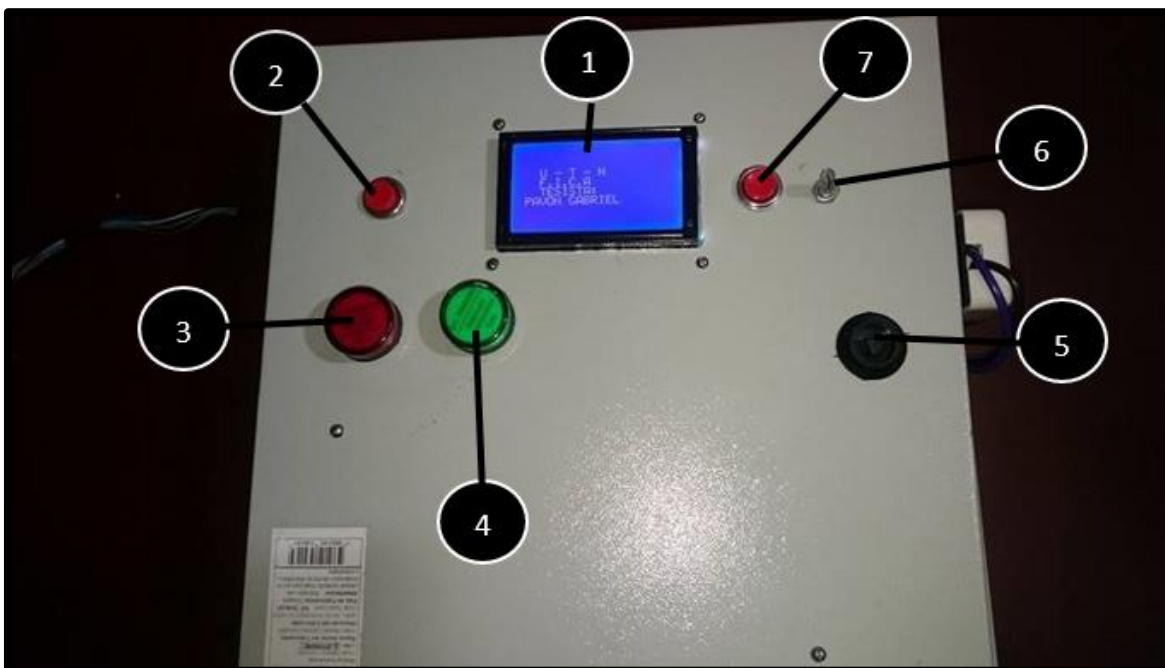


Figura 61 Parte externa del gabinete eléctrico
Fuente: Autor

1. Pantalla grafica GLCD 128x64
2. Pulsador de activación del sistema.
3. Luz piloto de retiro automático de pezoneras.
4. Luz piloto de Encendido del sistema.
5. Llave de Acceso.
6. Regulador de intensidad de la pantalla GLCD.
7. Pulsador de apagado de todo el sistema.

4.2 PRUEBAS DEL FUNCIONAMIENTO DEL SISTEMA.

4.2.1 Pruebas de hardware.

Las pruebas realizadas en el presente proyecto fueron establecidas de acuerdo a los alcances planteados.

Para evidenciar todo el proceso de hardware se procederá a especificar de manera detallada:

- En la pantalla grafica GLCD se visualiza un mensaje de bienvenida al encenderse todo el sistema.



Figura 62 Mensaje de bienvenida en GLCD
Fuente: Autor

- El operador es el encargado de acercar el lector RFID al tag o etiqueta a una distancia recomendada de 2 cm, el tag está situado en el arete de la oreja del bovino. En la pantalla GLCD se visualiza un mensaje de Identificación correcta o Identificación Incorrecta.



Figura 63 Mensaje de “Identificación correcta” del bovino en GLCD
Fuente: Autor

- Posteriormente el operador presiona el pulsador, para activar el sistema y el motor eléctrico para colocar las pezoneras. Si el pulsador es activado se enciende un led blanco de la placa principal. Y en la pantalla GLCD se visualiza un tiempo de espera determinado para que el operador pueda colocar las pezoneras en la ubre del animal.

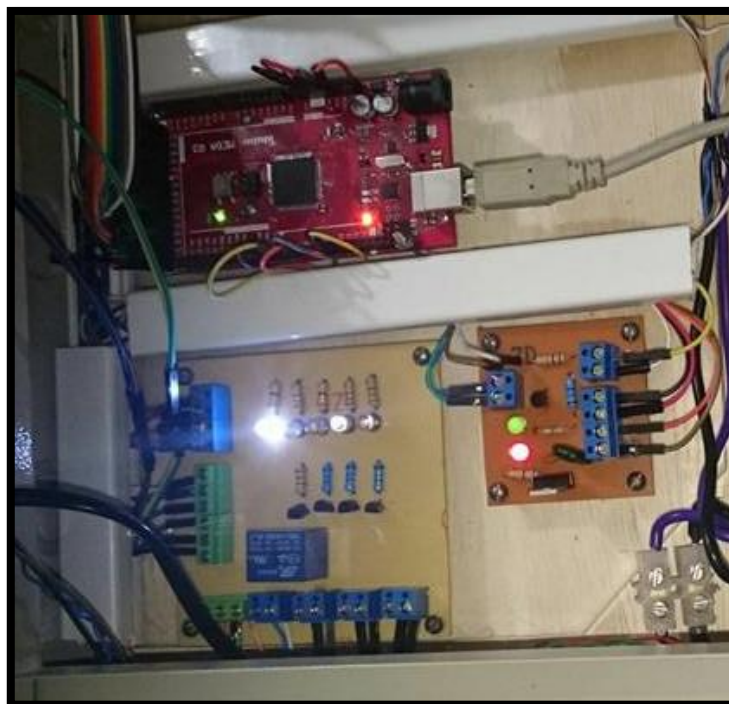


Figura 64 Led de visualización de pulsador activo
Fuente: Autor



Figura 65 Tiempo de espera en GLCD para colocar pezoneras
Fuente: Autor

- Después de haber terminado el tiempo de espera, se activa la luz piloto (color verde) de inicio del proceso y la válvula solenoide, con sus respectivos indicadores leds situados en la placa principal.



Figura 66 Luz piloto activado para inicio de proceso de extracción de leche
Fuente: Autor

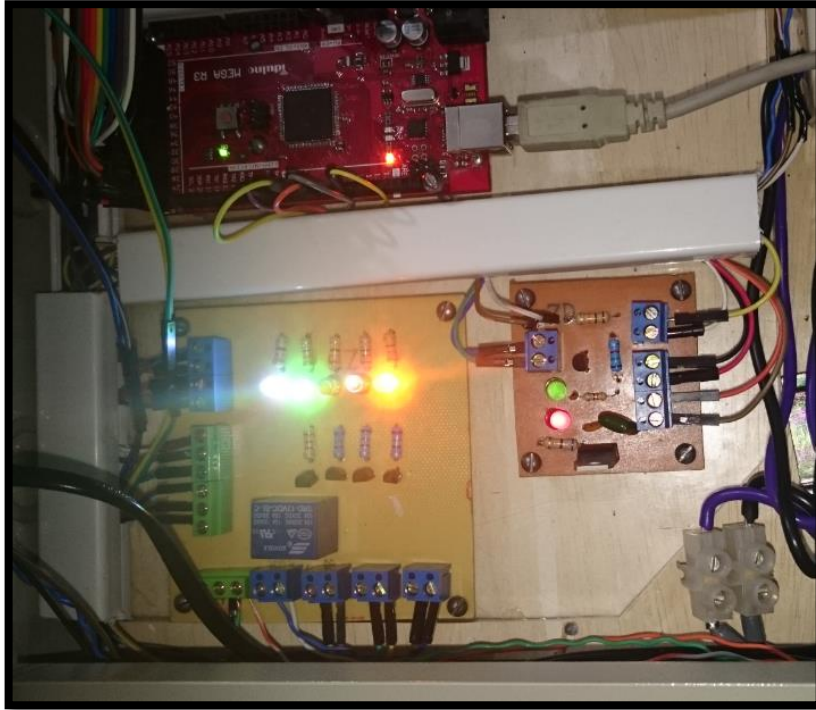


Figura 67 Leds de visualización de inicio de proceso y válvula solenoide activados
Fuente: Autor

- En el transcurso del proceso, se visualiza en la pantalla GLCD la cantidad de flujo que pasa por el sensor en L/min, el número de pulsaciones del sensor de flujo y el tiempo transcurrido en segundos.



Figura 68 Proceso de medición de datos en funcionamiento
Fuente: Autor

- Una vez que el sensor de flujo ya no tenga nada que medir, se terminará el proceso activándose la luz piloto de color rojo y activándose el motor eléctrico automáticamente para retirar las pezoneras, con sus respectivos leds indicadores. El microcontrolador Arduino envía la cantidad de leche producida en litros por el bovino a la minicomputadora.



Figura 69 Luz piloto activado para indicar fin de proceso de extracción de leche
Fuente: Autor

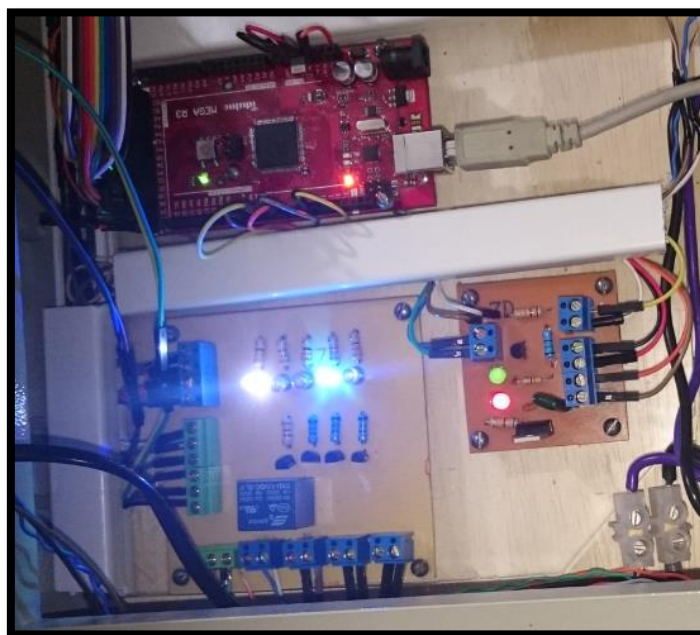


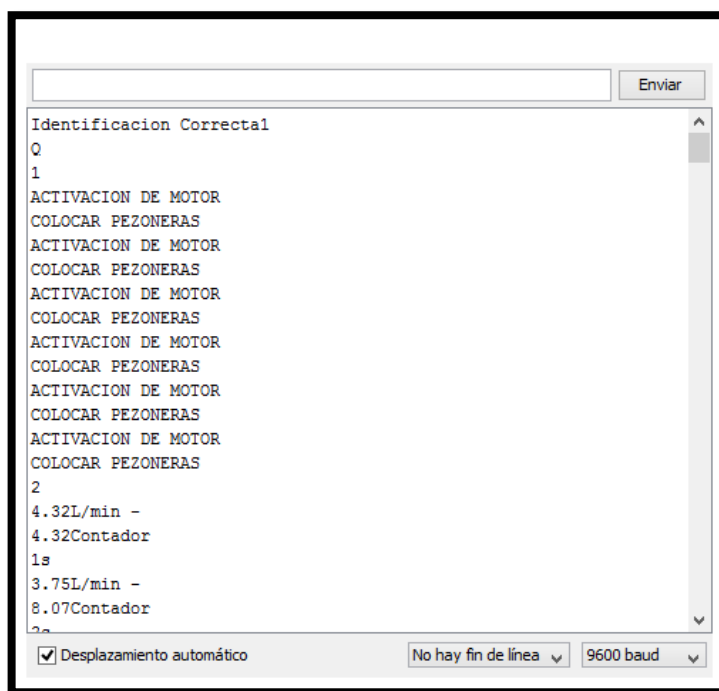
Figura 70 Leds de visualización de fin de proceso y motor eléctrico activados
Fuente: Autor

4.2.2 Pruebas de Software

Para realizar las pruebas de software en el microcontrolador se hizo uso del monitor de serie de la plataforma de Arduino, y para las pruebas de la minicomputadora se accedió remotamente por SSH (protocolo de Shell seguro) a través de un emulador de terminal como ZOC.

4.2.2.1 Pruebas realizadas en el microcontrolador Arduino Mega 2560

En la figura 71 se puede observar la letra Q, que corresponde a una etiqueta RFID cuando es identificada correctamente, el numero 1 corresponde cuando el pulsador es activado, estas dos variables son necesarias para activar el sistema y el motor eléctrico para colocar las pezoneras en la ubre del animal. Se puede observar en el terminal los mensajes de activación de motor.



```
Identificacion Correcta1
Q
1
ACTIVACION DE MOTOR
COLOCAR PEZONERAS
ACTIVACION DE MOTOR
COLOCAR PEZONERAS
ACTIVACION DE MOTOR
COLOCAR PEZONERAS
ACTIVACION DE MOTOR
COLOCAR PEZONERAS
ACTIVACION DE MOTOR
COLOCAR PEZONERAS
ACTIVACION DE MOTOR
COLOCAR PEZONERAS
2
4.32L/min -
4.32Contador
1s
3.75L/min -
8.07Contador
2s
```

Desplazamiento automático No hay fin de línea 9600 baud

Figura 71 Mensaje de “Identificación correcta” y “Activación de motor” en monitor serial
Fuente: Autor

Como se puede observar en la figura 72 del terminal, se escriben la cantidad de flujo en litros/minuto, el número de pulsaciones del sensor de flujo y el tiempo transcurrido.

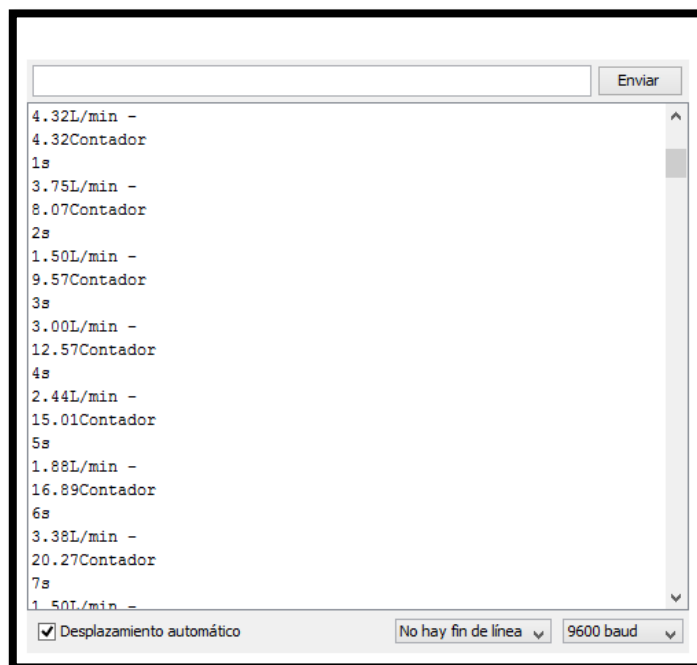


Figura 72 Mensajes del proceso de medición de datos en funcionamiento en monitor serial
Fuente: Autor

Para terminar el proceso se puede observar en la figura 73, que el sensor cuando ya no tenga que medir el flujo del caudal es 0.00L/min, seguidamente envía la cantidad de leche en litros que ha producido el bovino en ese tiempo transcurrido. Y activa el motor eléctrico para que se realice el retiro automático de las pezoneras.

```

Enviar
2.06L/min -
171.70Contador
101s
1.69L/min -
173.39Contador
102s
0.94L/min -
174.33Contador
103s
1.31L/min -
175.64Contador
104s
0.00L/min -
175.64Contador
105s
Z
1.69
2.93
ACTIVACION DE MOTOR RETIRO AUTOMATICO
FIN DE CICLO
 Desplazamiento automático
No hay fin de línea
9600 baud

```

Figura 73 Fin del proceso de extracción y envío de datos a la minicomputadora

Fuente: Autor

4.2.2.2 Pruebas realizadas en la minicomputadora Raspberry Pi B+ para la gestión de Base de Datos

Las pruebas fueron ejecutadas accediendo a la minicomputadora remotamente por SSH, y se obtuvo los siguientes resultados:

En el emulador del terminal se ejecuta de manera manual el script realizado con el lenguaje de programación de Python. Pero también se realizó la opción de que todos los scripts se ejecuten de manera automática al encenderse la Raspberry Pi B+. Esto se encuentra detallado en el manual del administrador.

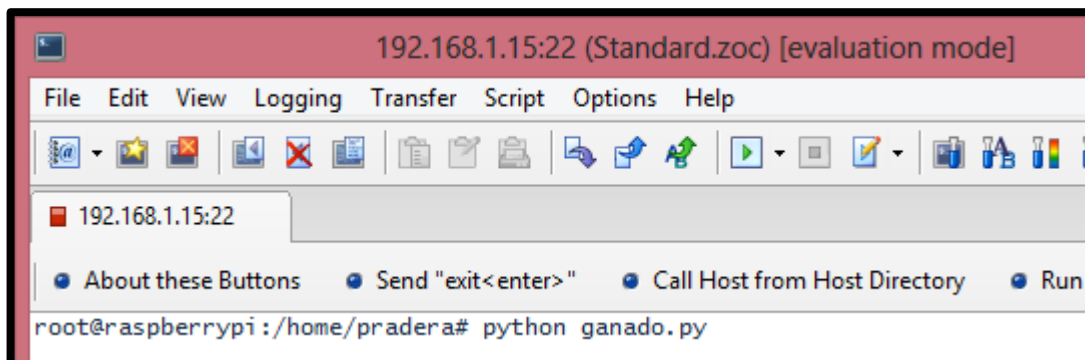


Figura 74 Ejecución manual del script del sistema
Fuente: Autor

Nos imprime en el terminal un mensaje de VACA1, el nombre del bovino a la cual se están almacenando los datos. Además se puede observar que para evidenciar dicho proceso se imprime en pantalla la medición que se obtuvo del microcontrolador Arduino, la fecha, la hora, el número de ordeño y un identificador de manera única de cada operación.

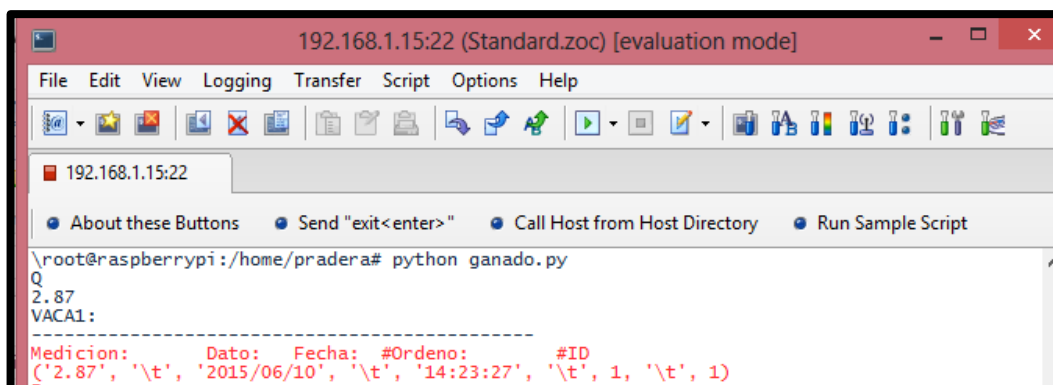


Figura 75 Impresión en consola de los datos obtenidos del microcontrolador
Fuente: Autor

Para validar que los datos se han almacenado correctamente, hacemos uso de la herramienta PhpMyAdmin, la cual nos permite gestionar la base de datos a través de una interfaz Web. Basta con escribir la dirección IP en un navegador Web, en este caso es 192.168.1.15/phpmyadmin, para poder acceder a la base de datos nos pedirá el nombre de usuario y la contraseña correspondiente.

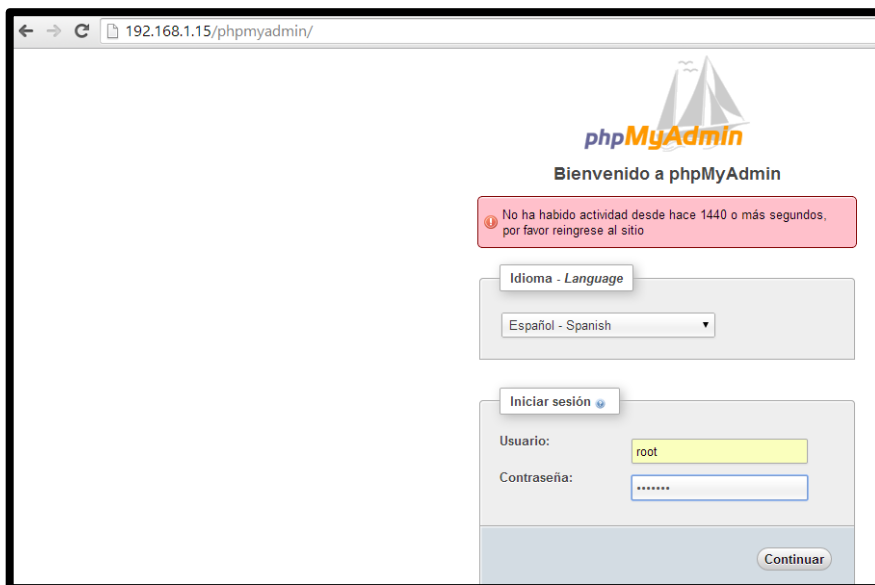


Figura 76 Gestión de base de datos por interfaz web
Fuente: Autor

En la figura 77 se observa que se accedió correctamente a la base de datos denominada Ganado, para verificar los resultados se da clic en el nombre de la tabla que corresponda al bovino, en este caso es Vaca1. Se puede observar que los datos han sido almacenados correctamente.

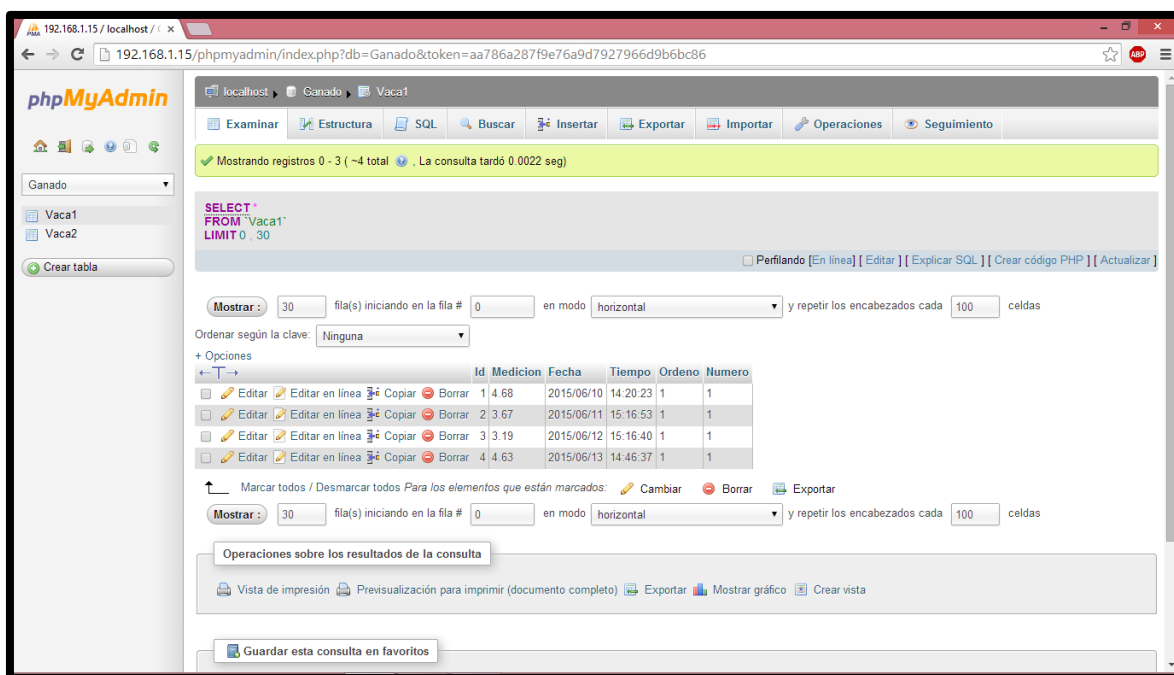


Figura 77 Datos almacenados en la base de datos
Fuente: Autor

Si se desea exportar los datos de cada bovino, se da clic en la pestaña Exportar y se selecciona el tipo de formato que se desea, puede ser Pdf, Excel, SQL, etc.

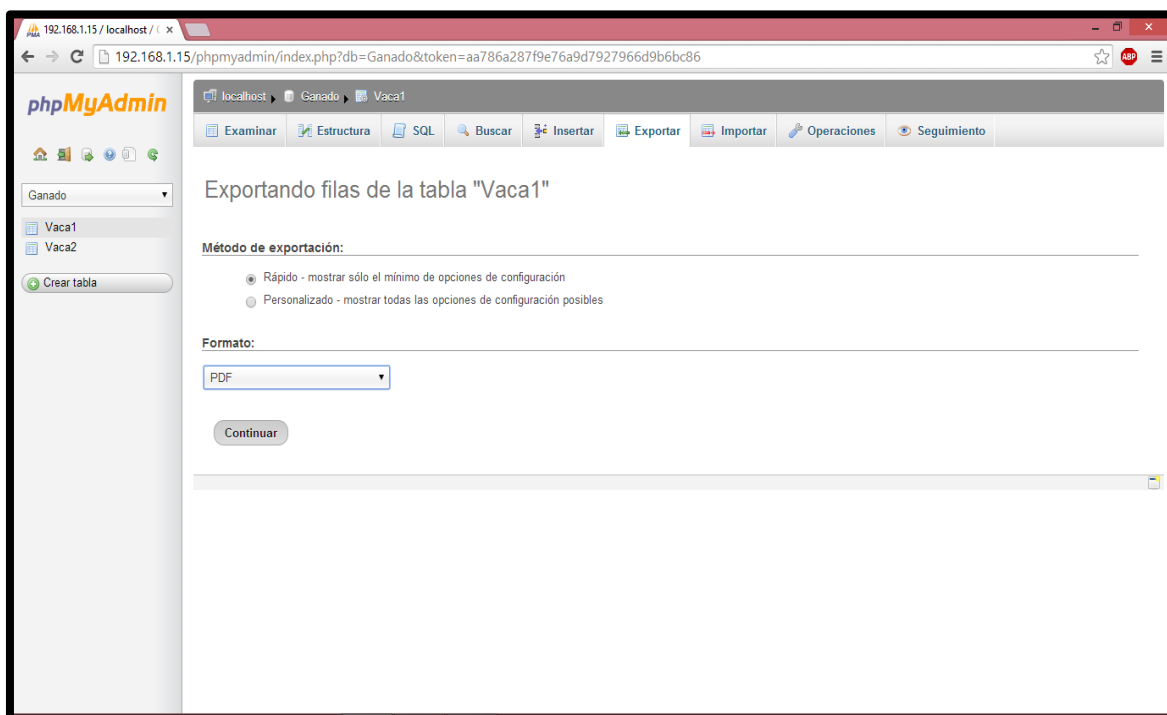


Figura 78 Exportación de los datos
Fuente: Autor

Comprobamos en la figura 79 y 80 que los datos del bovino han sido exportados de manera correcta en formato pdf y Excel, respectivamente.

Base de datos: Ganado, Tabla: Vaca1					
Id	Medicion	Fecha	Tiempo	Ordeno	Numero
1	4.68	2015/06/10	14:20:23	1	1
2	3.67	2015/06/11	15:16:53	1	1
3	3.19	2015/06/12	15:16:40	1	1
30	0.63	2015/06/13	14:46:37	1	1

Figura 79 Datos en formato pdf
Fuente: Autor

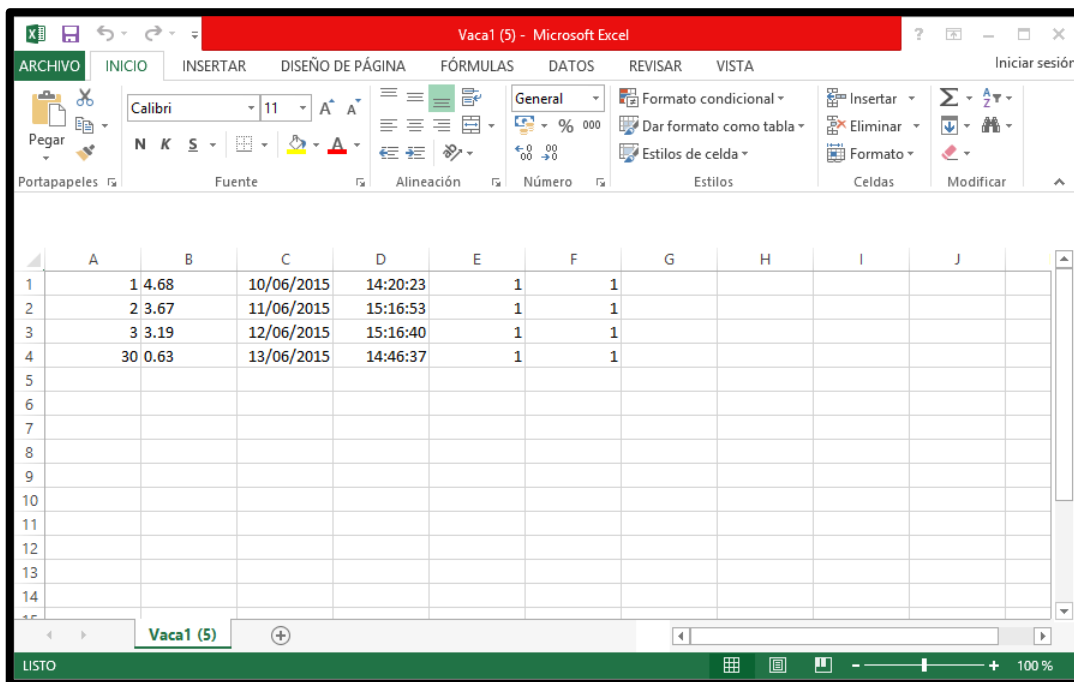


Figura 80 Datos en formato Word
Fuente: Autor

4.2.2.3 Resultados de almacenamiento de los datos en un archivo de texto en la minicomputadora

En la figura 81 se observa que se hace uso del editor de texto nano, para verificar que los datos se han almacenado en un archivo de texto.

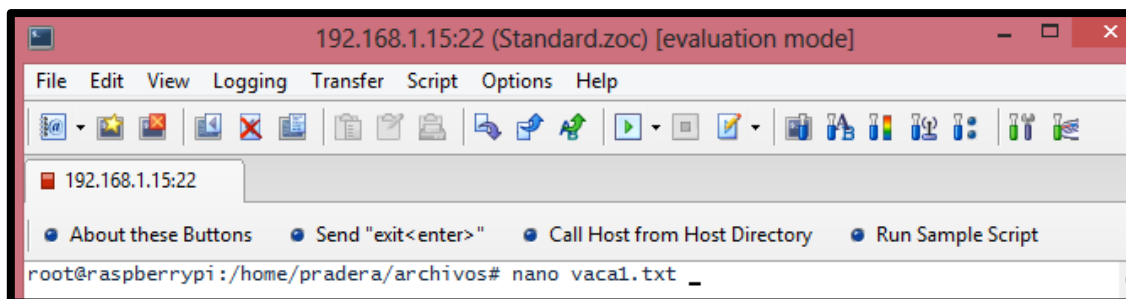
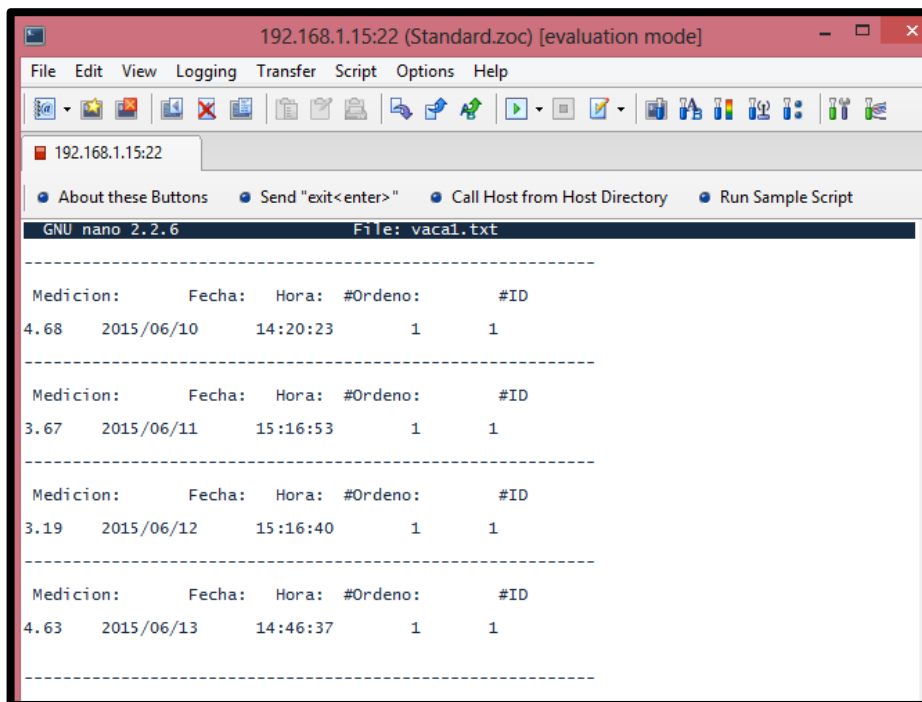


Figura 81 Ejecución del archivo de texto
Fuente: Autor

La figura 82 indica que los datos han sido almacenados correctamente en un archivo de texto.



```

192.168.1.15:22 (Standard.zoc) [evaluation mode]
File Edit View Logging Transfer Script Options Help
192.168.1.15:22
About these Buttons Send "exit<enter>" Call Host from Host Directory Run Sample Script
GNU nano 2.2.6 File: vacal.txt
-----
Medicion:      Fecha:  Hora:  #Ordeno:  #ID
4.68   2015/06/10   14:20:23   1      1
-----
Medicion:      Fecha:  Hora:  #Ordeno:  #ID
3.67   2015/06/11   15:16:53   1      1
-----
Medicion:      Fecha:  Hora:  #Ordeno:  #ID
3.19   2015/06/12   15:16:40   1      1
-----
Medicion:      Fecha:  Hora:  #Ordeno:  #ID
4.63   2015/06/13   14:46:37   1      1
-----

```

Figura 82 Verificación de datos almacenados en archivo de texto

Fuente: Autor

4.2.2.4 Resultados de acceso y de transferencia de la información a través de un servidor FTP

Como servidor FTP se instaló el daemon vsftpd en la minicomputadora, y como cliente FTP se utilizó el programa WinSCP por su facilidad de uso.

En la figura 83 se puede observar que para acceder al servidor FTP, se necesita de un nombre de usuario y su respectiva contraseña, el número de puerto en este caso es 21. Además en la parte del Servidor se debe ingresar la dirección IP o nombre del equipo.

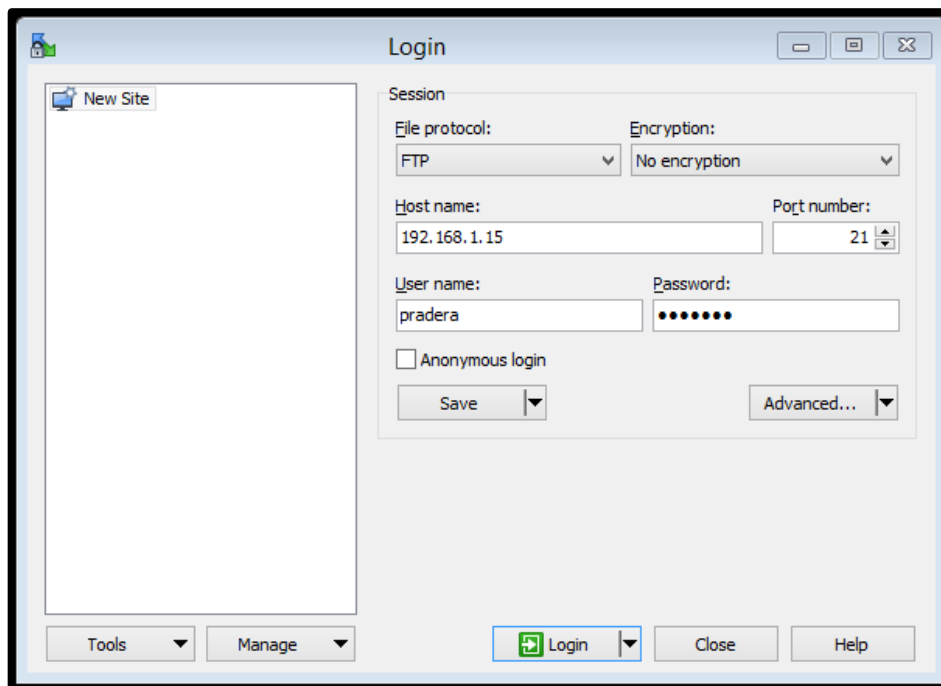


Figura 83 Conexión al servidor FTP

Fuente: Autor

Una vez que se ha conectado correctamente a la minicomputadora, se selecciona el archivo de texto o cualquier archivo que se desea transferir del sitio remoto, y simplemente para que se realice la copia al sitio local, se suelta el archivo en la carpeta que se desea del sitio local.

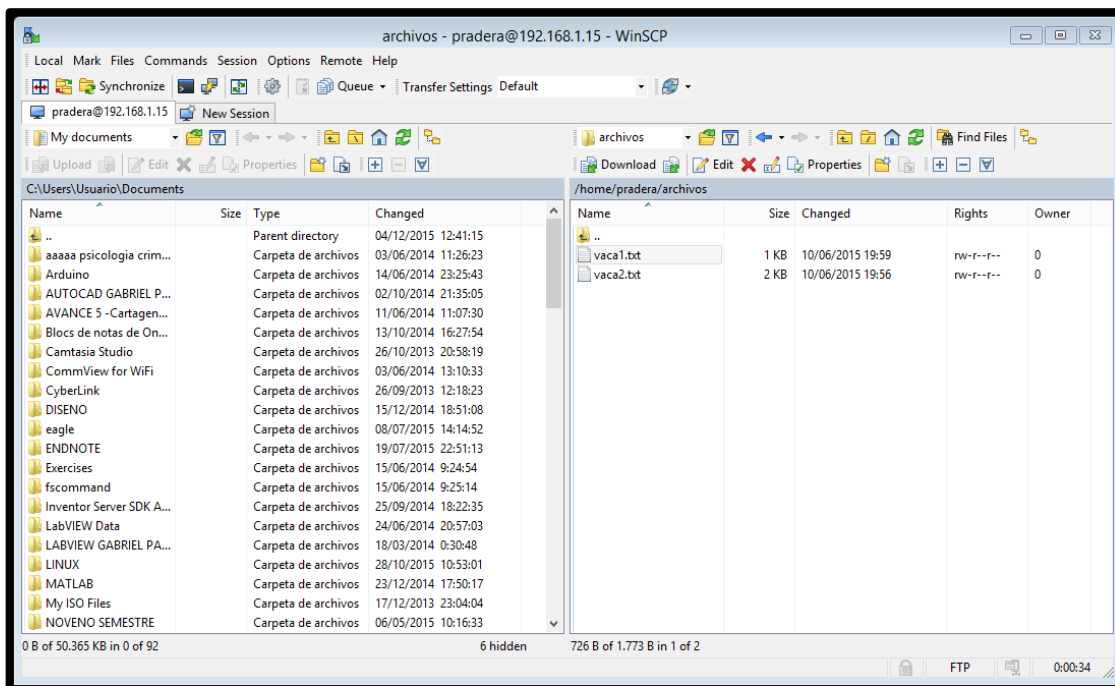


Figura 84 Selección de archivo que se desea transferir
Fuente: Autor

Se puede observar en la figura 85 que la transferencia del archivo de texto vacca1.txt se realizó con éxito desde la Raspberry Pi B+ hacia los documentos del cliente.

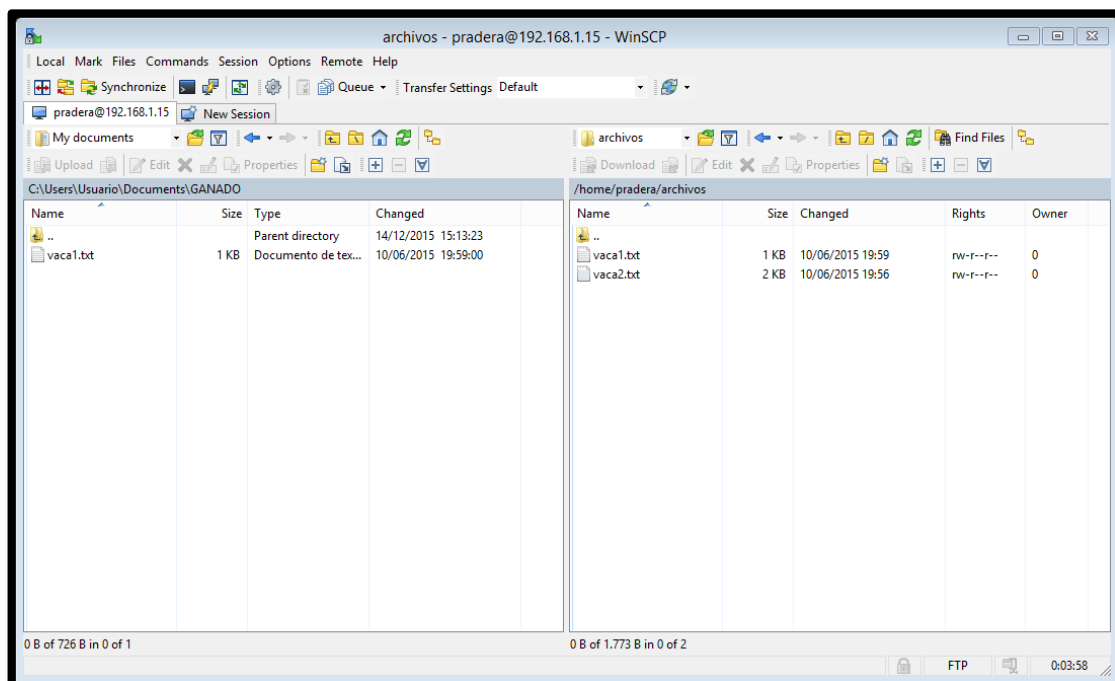


Figura 85 Transferencia correcta de archivo
Fuente: Autor

Finalmente se abre el archivo para evidenciar que la copia se realizó con éxito, sin ningún inconveniente.

```

1 -----
2
3 Medicion:      Fecha:      Hora:      #Ordeno:      #ID
4
5 4.68      2015/06/10      14:20:23      1      1
6
7 -----
8
9 Medicion:      Fecha:      Hora:      #Ordeno:      #ID
10
11 3.67      2015/06/11      15:16:53      1      1
12
13 -----
14
15 Medicion:      Fecha:      Hora:      #Ordeno:      #ID
16
17 3.19      2015/06/12      15:16:40      1      1
18
19 -----
20
21 Medicion:      Fecha:      Hora:      #Ordeno:      #ID
22
23 4.63      2015/06/13      14:46:37      1      1
24
25 -----
26

```

Figura 86 Verificación de archivo en sitio local

Fuente: Autor

4.2.2.5 Verificación del servidor web para monitorear y consultar los resultados previos del ganado bovino.

En un navegador web colocamos la dirección de la página web principal, en este caso es <http://192.168.1.15/pradera/index.php>. Nos pedirá un nombre de usuario y su respectiva contraseña, con la finalidad de que nuestro servidor web contenga privacidad.

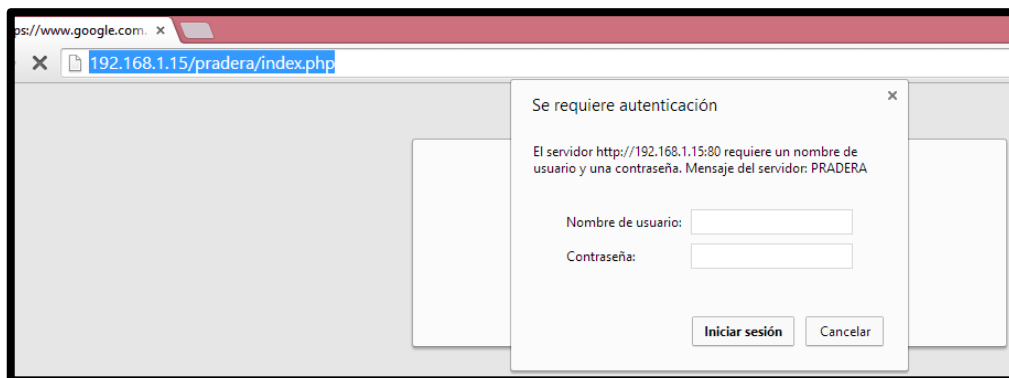


Figura 87 Autenticación en el servidor Web
Fuente: Autor

Una vez hayamos colocado el nombre de usuario y la contraseña correctamente, nos aparecerá las opciones de las tablas de los bovinos que se desea consultar.



Figura 88 Consultar resultados de cada bovino en servidor Web
Fuente: Autor

Se elige una opción y se da clic en Consultar, nos aparecerán los resultados respectivos del bovino seleccionado.

La Pradera UTN

Gabriel Pavón

DESCRIPCION DE MARIA

ID	Medicion	Fecha	Tiempo	Ordeno	Numero
1	4.68	2015/06/10	14:20:23	1	1
2	3.67	2015/06/11	15:16:53	1	1
3	3.19	2015/06/12	15:16:40	1	1
4	4.63	2015/06/13	14:46:37	1	1

Inicio

Diseñado por Gabriel Pavón - La pradera Utn

Figura 89 Verificación de los resultados del bovino seleccionado en un navegador Web
Fuente: Autor

Para regresar a la página web principal y consultar los resultados de otro bovino, damos clic en la opción de Inicio.

4.2.2.6 Verificación del envío del archivo con la información del bovino a través de un correo electrónico.

Para verificar que se ha enviado correctamente los archivos de cada bovino al correo electrónico, abrimos el servicio de correo Hotmail y comprobamos. Cabe mencionar que el correo destinatario puede ser de cualquier cuenta de correo electrónico y que el remitente es del servicio de correo de Gmail.

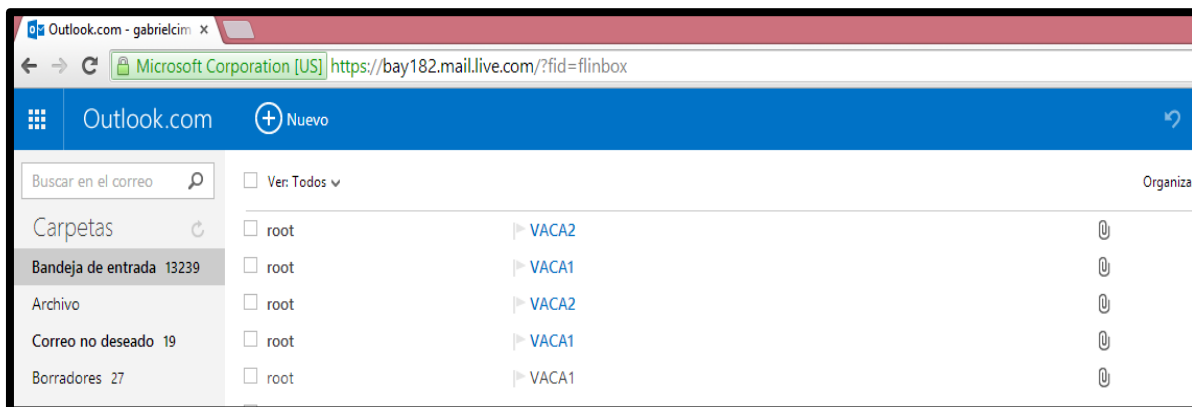


Figura 90 Verificación de envío del archivo por correo electrónico
Fuente: Autor

4.3 ANÁLISIS DE RESULTADOS

Primeramente para la obtención de la cantidad de leche que produce cada bovino, se procedió a realizar pruebas, para obtener la cantidad de flujo (caudal) en un minuto de manera empírica del sensor de flujo. Este resultado es fundamental para obtener los resultados lo más preciso posible.

Una vez obtenido esa cantidad el microcontrolador es el encargado de tomar muestras de la cantidad de flujo cada segundo, cuando se haya terminado el proceso, el microcontrolador realiza un promedio y transforma ese promedio obtenido en litros en el tiempo transcurrido de las muestras. En el Anexo 04 se muestran los cálculos para este proceso.

En la tabla 17 se exponen los resultados obtenidos del sensor de flujo para ver la cantidad de leche producida en litros de un bovino.

Tabla 17 *Resultados obtenidos del sensor de flujo*

ID	Fecha	Hora	# Ordeño	Medición del sensor	Porcentaje de
				(Litros)	Error %
1	10/06/2015	14:20:23	1	4.68 L	4
2	10/06/2015	14:31:21	1	4.53 L	3
3	10/06/2015	14:45:02	1	4.94 L	4
4	10/06/2015	14:56:50	1	3.98 L	3

Fuente: Autor

Como se puede observar en la tabla 17 los resultados de porcentaje de error se deben a que la composición de la leche varía entre animales, por sus propiedades físico-químicas, además de que el sensor de flujo es sometido a vacío por la máquina de ordeño lo que produce espuma.

CAPÍTULO V

En este capítulo se va analizar los costos del proyecto y el costo total, en el mercado existe una gran variedad de elementos como los dispositivos de propósito general, sensores y actuadores con los que se puede trabajar en este sistema. Para el desarrollo de este sistema se optó por seleccionar componentes que sean de fácil adquisición y bajo costo.

5.1 COSTOS DEL SISTEMA

A continuación se presenta una tabla con los costos correspondientes a la elaboración del proyecto, el cual puede aumentar en cuanto se requiera cubrir un mayor número de expectativas, colocando otro tipo de sensores o actuadores que tengan mayor precisión.

5.1.1 Costos del diseño y elaboración del prototipo.

En la tabla 18 se muestran los costos de los materiales utilizados para el desarrollo del proyecto.

Tabla 18 *Costo del diseño y elaboración del sistema*

ITEM	DESCRIPCIÓN	CANTIDAD	COSTO UNITARIO USD	COSTO TOTAL USD
1	Microcontrolador Arduino Mega 2560	1	60	60
2	Minicomputadora Raspberry Pi B+	1	100	100

3	Módulo RFID RC522	1	30	30
4	Módulo RTC 3231	1	20	20
5	Etiquetas RFID	15	4,5	67,5
6	Sensor de Flujo FS300A	1	25	25
7	Válvula Solenoide	1	23	23
8	Pantalla Grafica GLCD 128X64	1	30	30
9	Motor Plumas Hyundai HD72	1	35	35
10	Relés electromecánicos 24V	2	20	40
11	Porta relés	2	5	10
12	Relés 12V	1	2	2
13	Fuente de poder 24V	1	35	35
14	Breaker	1	15	15
15	Porta fusibles	1	10	10
16	Canaleta en metros	3	0,85	2,55
17	Gabinete Eléctrico	1	30	30
18	Luces Piloto	2	4	8
19	Resistencias	11	0,05	0,55
20	Diodos 4007	3	0,2	0,6
21	Potenciómetros	1	0,25	0,25
22	Cables UTP cat6 a (m)	40	0,8	32
23	Borneras de 2 pines	10	0,4	4
24	Bornera de 3 pines	3	0,5	1,5
25	Estaño en metros	3	0,6	1,8
26	Pulsador	2	1	2
27	Acido para baquelitas	2	0,6	1,2
28	Regulador de Voltaje 7812	1	1,2	1,2
29	Disipador de calor	1	0,8	0,8
30	Capacitor cerámicos	2	0,2	0,4
31	Transistores 3904	5	0,3	1,5
32	Leds	7	0,15	1,05
33	Placa de Acrilico	1	8	8
34	Alambre de Acero(m)	10	1,5	15
35	Brocas de Acero	3	7	21
36	Barra sin fin	1	7	7
37	Barras de Silicona	5	1,5	7,5
38	Cable AWG #2x12(m)	5	0,8	4
39	Cable AWG # 16(m)	12	0,8	9,6
40	Carrete de acero inoxidable	1	40	40

41	Lamina de Acero inoxidable (espesor 1mm)	1	20	20
42	Amarraderas	1	3	3
43	Bushing	4	0,8	3,2
			TOTAL=	730,2

Fuente: Autor

5.2 CONCLUSIONES

- Al culminar el desarrollo del proyecto se pudo observar que se cumplieron con los objetivos planteados, en su totalidad. Además, mejoró su eficiencia y rendimiento del ordeño. Debido a que el tiempo de ordeño del ganado bovino se lo realizaba en aproximadamente 1 hora y media, y ahora con el sistema se lo realiza en aproximadamente 1 hora.
- El sistema tiene como particularidad que puede ser adaptado a cualquier necesidad que posea el establo del ganado bovino, porque tanto el hardware como el software del Arduino Mega 2560 y de la Raspberry Pi B+ son flexibles. Al módulo Arduino se lo pueden conectar hasta 6 sensores de flujo (funciones especiales interrupción), 4 minicomputadoras (UART) y conectar múltiples módulos Rfid (SPI). La minicomputadora puede ser conectada con 4 módulos Arduinos (UART).
- Para evitar fallas en la fecha y hora cuando la Raspberry Pi B+ no pueda conectarse al servidor NTP (Network Time Protocol), se integró un módulo RTC ds3231 (Real Time Clock) el cual resuelve el problema y obtiene correctamente dichos parámetros en su sistema.
- El porcentaje de error obtenido del sensor de flujo se redujo a un 3-4%, debido a que el microcontrolador toma muestras del flujo del caudal cada segundo, realiza una sumatoria y por último ejecuta un promedio de las muestras. Además, el sensor de

flujo es sometido a vacío por la máquina de ordeño lo que produce un poco de espuma y también afecta al porcentaje de error.

- Este sistema permite el registro automático de la producción de leche del ganado bovino, con un control fiable de aproximadamente 108 litros de leche por día. Este registro se realizaba manualmente en una hoja de papel, sin ningún tipo de control, esta acción se presta a muchos inconvenientes.

- El sistema para garantizar la seguridad de los datos obtenidos es almacenado en una base de datos, en un archivo de texto, enviado al correo electrónico del administrador, tiene habilitado un servidor FTP para acceder y transferir la información del ganado bovino en cualquier instante. Y así poder tener respaldo de la información.

5.3 RECOMENDACIONES

- Es necesario que el gabinete o tablero eléctrico no sea manipulado por cualquier persona sin experiencia, ya que puede causar algún factor que pueda afectar al funcionamiento del sistema.
- Se recomienda que el sistema completo tenga por lo menos dos revisiones en el año, para garantizar el buen funcionamiento. Ya que al tratarse de dispositivos electrónicos pueden sufrir algún fallo. Aunque el sistema consta con un subsistema de protección para evitar daños a los dispositivos electrónicos.
- Las etiquetas RFID deben ser revisadas diariamente que no contengan ni un daño, ya que están expuestas a sufrir golpes al estar situadas en los aretes de los bovinos. Son animales de gran tamaño, con cuerpo robusto y gran peso.
- Debido a que el sistema se encuentra situado en un ambiente libre sin seguridades físicas, como puertas, paredes, etc. Debe recibir protección adicional, como un sistema de cámaras de vigilancia, para estar protegidos contra el vandalismo y la manipulación del sistema.
- Si se desea que el servidor web resuelva su dirección IP por un nombre de dominio, se puede hacer uso de un DNS dinámico como NO-IP.
- Además debido al tamaño de la memoria de la micro SD es reducida en comparación con otros ordenadores, lo que se recomienda para tener mayor almacenamiento es

instalar iSCSI y así conectarse con un servidor para obtener mayor almacenamiento virtual.

- Se recomienda instalar en la Raspberry Pi B+ algún sistema de gestor de contenidos web como Joomla o WordPress, y así tener una página web más dinámica.

REFERENCIAS BIBLIOGRÁFICAS:

TEXTOS:

- ACHA, S. C., Manuel; PÉREZ, Julio; RIOSERAS, Miguel. Electrónica Digital Introducción a la Lógica Digital Teoría, Problemas y Simulación: Alfaomega Rama.
- AETIC, O. Y. La tecnología RFID: Usos y oportunidades.
- BOLTON, W. (2006). Sistemas de Control Electrónico en la Ingeniería Mecánica y Eléctrica (Tercera ed.): Alfaomega.
- CISCO. Principios básicos de enrutamiento y switching. CCNA1 V5.
- COLLAGUAZO, G. Sistemas Microprocesados.
- CREUS, A. (2011). Instrumentación Industrial (Octava ed.): Alfaomega.
- David, P. V. H. T. D. B. J. Análisis y diseño de circuitos lógicos digitales (Primera ed.): Prentice-Hall.
- ENRIQUEZ, R. (2009). Guía de Usuario de Arduino.
- KUMAR, S. (2010). Introducción a la Robótica: Mc Graw Hill.
- MATURANA, C. (2006). RFID: El código de barras inteligente para Bibliotecas. Chile.
- PALLAS, R. Sensores y Acondicionadores de señal (Cuarta ed.): Marcombo.
- PETERSEN, R. (2009). LINUX Manual de Referencia (Sexta ed.): Mc Graw Hill.
- Robótica, T. d. (2012). Libro Básico de Arduino (Primera ed.). Bogotá, Colombia.
- SÁNCHEZ, J. (2012). Servidores de Aplicaciones Web.
- STEREN. ABC de la Mecatrónica.
- TELETRONICA. Introducción a la identificación por radio frecuencia (Primera ed.).

TESIS:

- ARMENDARIZ, J. C. (2009). Análisis comparativo de las tecnologías RFID, HID y AWID para proponer un sistema para la ubicación y seguridad de los libros en la biblioteca de la FIEE de la E.P.N. Escuela Politécnica Nacional, Quito-Ecuador. Retrieved from <http://bibdigital.epn.edu.ec/bitstream/15000/1085/1/CD-1923.pdf>
- BAQUE, G. V., José (2011). INTRANET basada en plataforma LAMP con integración de dispositivos móviles que permita el acceso a los datos de los distintos departamentos de asistencia médica del UPOCAM. Universidad Laica "Eloy Alfaro" de Manabí, Manta-Ecuador.
- SILVA, A. B., Julio (2013). Activación de Alarmas Remotas mediante WIFI entre minicomputadoras Raspberry PI en aplicaciones de Seguridad. Escuela Superior Politécnica del Litoral, Guayaquil-Ecuador.

PAGINAS WEB:

- Arduino. Arduino Due. <https://www.arduino.cc/en/Main/ArduinoBoardDue>
- Arduino. Arduino Leonardo.
<https://www.arduino.cc/en/Main/ArduinoBoardLeonardo>
- Arduino. Arduino Mega 2560.
<https://www.arduino.cc/en/Main/ArduinoBoardMega2560>
- Arduino. Arduino Uno. <https://www.arduino.cc/en/Main/ArduinoBoardUno>
- Arduino. Intel Galileo. <https://www.arduino.cc/en/ArduinoCertified/IntelGalileo>
- Arduino. Tabla comparativa de Arduinos.
<https://www.arduino.cc/en/Products.Compare>

- ASANZA, F. PcDuino o la simbiosis del miniPC y Arduino.
<http://minimaxpc.es/htm/ecoDui01.html>
- INTEL. Intel Galileo. <http://www.intel.la/content/www/xl/es/do-it-yourself/galileo-maker-quark-board.html>
- Isaac, P. Raspberry Pi vs pcDuino: análisis de miniPCs.
<http://architecnologia.blogspot.com/2013/03/raspberry-pi-vs-analisis-de-minipcs.html>
- LinkSprite, T. PcDuino. <http://www.pcdduino.com/get-sarted-with-pcdduino-introduction-to-pcdduino/>
- LinkSprite, T. PcDuino Lite. http://www.linksprite.com/?page_id=803
- MINA, G. (2015). Robótica Embebida.
https://eva.fing.edu.uy/pluginfile.php/88551/mod_resource/content/1/Introducci%C3%B3n%20a%20la%20IA%20y%20rob%C3%B3tica.pdf
- RaspberryPI. Que es un Raspberry PI. <https://www.raspberrypi.org/help/what-is-a-raspberry-pi/>
- Ultra-lab. La ola de los Mini-PCs. <http://ultra-lab.net/blog/la-ola-de-los-mini-pcs>

ANEXO 01: FIRMWARE DEL MICROCONTROLADOR ARDUINO MEGA 2560

Este código hace referencia a un solo bovino.

```
// Incluye libreria RFID, SPI
```

```
#include <SPI.h>
```

```
#include <RFID.h>
```

```
#define SS_PIN 53
```

```
#define RST_PIN 9
```

```
RFID rfid(SS_PIN, RST_PIN);
```

```
// Incluye libreria de GLCD
```

```
#include <glcd.h>
```

```
// Incluye libreria de fuentes
```

```
#include <fonts/allFonts.h>
```

```
// Setea variables:
```

```
int serNum0;
```

```
int serNum1;
```

```
int serNum2;
```

```
int serNum3;
```

```
int serNum4;
```

```
//Codigo de una TAG
```

```
int Num0 = 9;
```

```
int Num1 = 47;
```

```
int Num2 = 130;
```

```
int Num3 = 74;
```

```
int Num4 = 238;
```

```
//VARIABLES GLOBALES
```

```
int x=0,y=0,z=0,a=0;
```

```
char var[]={ 'Q','P'};
```

```
//VARIABLES DE BLOQUE 1
```

```
const int buttonPin_1 = 4;
```

```
int buttonState_1 = 0;
```

```
int ledPin_button = 38;
```

```
int led_estado_1 = 6;
```

```
int led_fin_ciclo_1 = 44;
```

```
int valvula_1 = 40;
```

```
int motor_1 = 42;
```

```
int motor_2 = 7;
```

```
//VARIABLES DE SENSOR DE FLUJO
```

```
int i=0;
```

```
float media=0;
```

```
float total = 0;
```

```
float promedio=0;
```

```
float contaPulso;
```

```
float vazao;
```

```
//Subrutina de Interrupcion
```

```
void rpm ()
```

```
{
```

```
    contaPulso++;
```

```
}
```

```
//Subrutina de lectura del TAG
```

```
void lecturaRFID()
```

```
{
```

```
    if (rfid.isCard()) { //Identifica si es una tag
```

```
        if (rfid.readCardSerial()) { //Lee la tag
```

```
            if (rfid.serNum[0] != serNum0
```

```
                && rfid.serNum[1] != serNum1
```

```
                && rfid.serNum[2] != serNum2
```

```
                && rfid.serNum[3] != serNum3
```

```
                && rfid.serNum[4] != serNum4
```

```
            ) {
```

```
                serNum0 = rfid.serNum[0]; //Almacenan la lectura del código de la tag
```

```
                serNum1 = rfid.serNum[1];
```

```
                serNum2 = rfid.serNum[2];
```

```
                serNum3 = rfid.serNum[3];
```

```
                serNum4 = rfid.serNum[4];
```

```
//Compara la tag leída con el código

if (rfid.serNum[0] == Num0 && rfid.serNum[1] == Num1 && rfid.serNum[2]
== Num2 && rfid.serNum[3] == Num3 && rfid.serNum[4] == Num4) {

    delay(100);

    GLCD.print("Identificacion Correcta1");

    GLCD.CursorTo(0, 2);

    GLCD.println("VACA1");

    delay(2000);

    GLCD.ClearScreen(); //Borra pantalla de GLCD

    delay(100);

    y=1; //Variable de comparación para saber que tag es

    }

}

else {

    //Imprime mensaje en GLCD si la tarjeta no es correcta

    GLCD.print("IDENTIFICACION INCORRECTA");

    delay(1000);

    GLCD.ClearScreen();

    }

}

}

}
```

```
//Subrutina de Vaca Isabella
```

```
void Isabella()
```

```
{
```

```
if (rfid.serNum[0] == Num0 && rfid.serNum[1] == Num1 && rfid.serNum[2] == Num2
&& rfid.serNum[3] == Num3 && rfid.serNum[4] == Num4) {
```

```
if (y==1 ) //Variable de comparación
```

```
{
```

```
if (a==1) //Variable si el pulsador se activado
```

```
{
```

```
for (int t=0; t<=30; t++) //Encender motor
```

```
{
```

```
/////Serial.println ("ACTIVACION DE MOTOR");
```

```
/////Serial.println ("COLOCAR PEZONERAS");
```

```
digitalWrite(motor_1, HIGH);
```

```
digitalWrite(motor_2, LOW);
```

```
delay(1000);
```

```
}
```

```
for (int l=0; l<=9; l++) //Tiempo de espera para colocar pezoneras
```

```
{
```

```
GLCD.CursorTo(7, 1);
```



```
GLCD.print("ESPERE 9 SEG.");

GLCD.println("");

GLCD.CursorTo(7, 3);

GLCD.print("SEG.");

GLCD.println("");

GLCD.CursorTo(5, 3);

GLCD.print(1);

GLCD.println("");

delay(1000);

digitalWrite(motor_1, LOW);

digitalWrite(motor_2, LOW);

}

GLCD.ClearScreen(); //Borra la pantalla del GLCD

y++;

a++;

int contador=420; //Variable de tiempo máximo de ordeno

while (contador != 0){

digitalWrite(led_estado_1, HIGH); //Enciende led On y válvula solenoide

digitalWrite(valvula_1, HIGH);

digitalWrite(led_fin_ciclo_1, LOW);
```

```
digitalWrite(motor_1, LOW);  
  
digitalWrite(motor_2, LOW);  
  
contador = contador - 1; //Decremento de la variable de tiempo máximo de ordeno  
  
contaPulso = 0;  
  
delay(900);  
  
vazao = (contaPulso / 5.329); //Ecuación para obtener los L/min  
  
media = media+vazao; //Numero de pulsaciones  
  
i++;  
  
  
delay(50);  
  
  
GLCD.CursorTo(0, 1); //Imprime en pantalla GLCD los resultados  
  
GLCD.print("L/min");  
  
GLCD.CursorTo(0, 2);  
  
GLCD.print(vazao);  
  
GLCD.println("");  
  
GLCD.CursorTo(0, 3);  
  
GLCD.print("Pulsos");  
  
GLCD.CursorTo(0, 4);  
  
GLCD.print(media);  
  
GLCD.println("");  
  
GLCD.CursorTo(0, 5);  
  
GLCD.print("Tiempo");
```

```
GLCD.CursorTo(0, 6);

GLCD.print(i);

GLCD.println("");

delay(50);

if (vazao <= 0.01 && y >= 2) //Ciclo de fin de sensado
{
    int time = 0;

    time = i - 1;

    promedio = media/time; //Variable para obtener el promedio L/min
    total = (promedio * time)/60; //Transformamos a L

    Serial.print(var[0]);Serial.print(","); //Enviamos datos a miniPC

    Serial.println(total);

    delay(500);

    GLCD.CursorTo(0, 7); //Imprime en pantalla GLCD el total de L

    GLCD.print(total);

    GLCD.println("");

    delay(2000);

    y=0;

    z=1;

    contador=0;
```

```
    fin_ciclo();    //Llamamos a subrutina de fin de ciclo
}

if (contador<=0)
{
    contador=0;
}
}
}

else
{
    y=0;
}
}
}

//Subrutina de fin de ciclo
void fin_ciclo()
{
if (z==1)
{
    /////Serial.println ("ACTIVACION DE MOTOR RETIRO AUTOMATICO");
```

```
/////Serial.println ("FIN DE CICLO");

z++;

for (int j =0 ; j<6; j++)

{

digitalWrite(led_estado_1, LOW); //Enciende led de OFF y motor para retiro

digitalWrite(valvula_1, LOW);

digitalWrite(led_fin_ciclo_1, HIGH);

digitalWrite(motor_1, LOW);

digitalWrite(motor_2, HIGH);

media = 0;

i = 0;

delay(1000);

}

}

}
```

//Subrutina de mensaje de Bienvenida

```
void GLCD_PP()

{

GLCD.CursorTo(3, 2);

GLCD.print("U - T - N");

GLCD.CursorTo(3, 3);
```

```
GLCD.print("F.I.C.A");
GLCD.println("\n");
GLCD.CursorTo(3, 4);
GLCD.print("TESISTA:");
GLCD.println("\n");
GLCD.CursorTo(1, 5);
GLCD.print("PAVON GABRIEL");
GLCD.println("");
}

//Subrutina de Activación del Pulsador
void boton_1()
{
  buttonState_1 = digitalRead(buttonPin_1);
  if (buttonState_1 == LOW) {
    a = 1;      //Variable de comparación
    digitalWrite(ledPin_button, HIGH);
    delay(200);
  }

  else {
    a = 0;
    digitalWrite(ledPin_button, LOW);
  }
}
```

```
}

//Función de inicializar los modos de trabajo

void setup()

{

  SPI.begin(); //Inicia comunicación SPI

  Serial.begin(9600); //Activa la comunicación serial

  attachInterrupt(0, rpm, RISING); //Habilita el recurso de interrupción

  rfid.init(); //Inicializa el módulo RFID

  GLCD.Init();//Inicia la pantalla GLCD

  GLCD.SelectFont(System5x7); //Tipo de fuente

  pinMode(buttonPin_1, INPUT); //Declaración de pines E/S

  pinMode(ledPin_button, OUTPUT);

  pinMode(led_estado_1, OUTPUT);

  pinMode(led_fin_ciclo_1, OUTPUT);

  pinMode(valvula_1, OUTPUT);

  pinMode(motor_1, OUTPUT);

  pinMode(motor_2, OUTPUT);

  GLCD_PP(); //LLamado a subrutina de mensaje de bienvenida

  delay(5000);
```

```
GLCD.ClearScreen();    //Borrado de la pantalla GLCD
}

//Ciclo infinito
void loop()
{
    digitalWrite(led_estado_1, LOW); //Apagamos todas las E/S
    digitalWrite(valvula_1, LOW);
    digitalWrite(led_fin_ciclo_1, LOW);
    digitalWrite(motor_1, LOW);
    digitalWrite(motor_2, LOW);
    digitalWrite(ledPin_button, LOW);

    boton_1(); //LLamado de subrutina de botonn
    lecturaRFID(); //LLamado de subrutina RFID
    isabella(); //LLamado de subrutina de Vacal
    rfid.halt(); //
}
```


ANEXO 02: FIRMWARE DEL SCRIPT EN RASPBERRY PI B+ MEDIANTE PYTHON

Este código hace referencia a un solo bovino, y se encarga del almacenamiento de información en base de datos y archivos de texto, así como también de enviar el archivo por correo electrónico. Se denomina el script ganado.py

```
#!/usr/bin/python
```

```
# -*- coding: utf-8 -*-
```

```
#Importacion de librerias
```

```
import MySQLdb as mdb
```

```
import time
```

```
import datetime
```

```
import sys
```

```
import os
```

```
import serial
```

```
now = datetime.datetime.now()
```

```
#Habilita Comunicación serial
```

```
arduino = serial.Serial('/dev/ttyACM0', baudrate=9600)
```

```
arduino.open()
```

```
#variable para saber las veces q ha sido ordenada
```

```
ordeno_1 = 0
```

```
ordeno_2 = 0
```

```
Num_veces1 = 0
```

```
Num_veces2 = 0
```

```
#Conexion con la base de datos
```

```
con = mdb.connect(host='localhost', user='pradera', passwd='pradera', db='Ganado')
```

```
cur = con.cursor()
```

```
#Creamos la tabla denominada Vaca1
```

```
#cur.execute("CREATE TABLE Vaca1 (Id INT PRIMARY KEY AUTO_INCREMENT,  
Medicion VARCHAR(100), Fecha VARCHAR(100), Tiempo VARCHAR(100), Ordeno  
VARCHAR(100), Numero VARCHAR(100))")
```

```
#Asignamos el tipo de formato de las variables en la base de datos
```

```
vaca1 = "INSERT INTO Vaca1(Medicion, Fecha, Tiempo, Ordeno, Numero)  
VALUES(%s, %s, %s, %s, %s)"
```

```
#GRADAR DATOS EN UN ARCHIVO
```

```
archivo = "/home/pradera/archivos/vaca1.txt"
```

```
archi = open(archivo,'a')
```

```
archi.write("\n-----\n")
```

```
archi.write("\n Medicion:    Dato: Fecha: #Ordeno:    #ID \n\r")
```

```
#Objeto para guardar los datos en el archivo de texto
```

```
def archivo1():  
  
    archi.write("\n")  
  
    archi.write(texto)  
  
    archi.write("\t")  
  
    archi.write(sample_date)  
  
    archi.write("\t")  
  
    archi.write(sample_time)  
  
    archi.write("\t")  
  
    archi.write(str(ordeno_1))  
  
    archi.write("\t")  
  
    archi.write(str(Num_veces1))  
  
    archi.write("\t\n\r")
```

#Ciclo Infinito

```
while 1:
```

```
    count = 10
```

```
    #Almacenamos los datos que son enviados desde el microcontrolador en variables
```

```
    line = arduino.readline().rstrip()
```

```
    letra,texto = line.split(",")
```

```
    time.sleep(0.1)
```

```
    print (letra)
```

```

print (texto)

#Comparamos la letra al bovino que le pertenezca
if letra == 'Q':
    ordeno_1 = ordeno_1+1
    print ('VACA1: ')
    Num_veces1 = Num_veces1+1

#variables de tiempo
sample_date = time.strftime("%Y/%m/%d")
sample_time = time.strftime ("%H:%M:%S")

#Imprime en pantalla
print "-----"
print chr(27)+"[1;31m"+"Medicion:   Dato: Fecha: #Ordeno:   #ID"
print (texto,"\t",sample_date,"\t",sample_time,"\t",ordeno_1,"\t",Num_veces1)

#Almacenamos las variables en la tabla Vaca1
cur.execute(logAnalogo, (texto, sample_date, sample_time, ordeno_1,
Num_veces1))

#Actualiza las tablas de la base de datos
con.commit()

time.sleep(0.1)

```

```
#Lama al objeto para grabar datos en archivo
archivo1()

#Cerramos el archivo de texto
archi.close()

#Tiempo de espera de 1 segundo
time.sleep(0.1)

#Enviamos el archivo por correo electrónico
os.system("sudo mpack -s 'VACA1' /home/pradera/archivos/vaca1.txt
gabrielcime@live.com")

# except KeyboardInterrupt:
#     print "\ndone"
```

ANEXO 03: FIRMWARE DEL SCRIPT EN RASPBERRY PI B+ PARA REVISAR INFORMACIÓN EN PAGINA WEB MEDIANTE PHP Y HTML

Script de la página principal: index.php

```
<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Pradera UTN</title>

</head>

<img src='cabecera.jpg' width=1400 height=350 vspace=3 hspace=0/>

<div align=center>

<body>

<form name="form" method="post" action="opciones.php">

<br />

<h2>Elija una opcion:</h2><select name="Animal">

<option value="vaca1">Vaca1</option>

<option value="vaca2">Vaca2</option>

</select>

<br>

<input type="submit" value="Consutar" title="Consultar">

</form>

<img src='header.jpg' width=480 height=480/>

</body>

</div>
```

```
<img src='foot.jpg' width=1400 height=50 vspace=3 hspace=0/>
</html>
```

Script de la selección de Opciones: opciones.php

```
<?php
$Animal = $_POST["Animal"];

if ($Animal == "vaca1")
{
    header('Location: http://192.168.1.15/pradera/vaca1.php');
}
if ($Animal == "vaca2")
{
    header('Location: http://192.168.1.15/pradera/vaca2.php');
}

?>
```

Script de la Consulta Vacal en la base de datos: vaca1.php

```
<html>
<head>
<img src='cbvaca.jpg' width=1400 height=350 vspace=3 hspace=0/>
<div align=center>
</head>
<title>vacal</title>
<body>
```

```
<h1>DESCRIPCION DE MARIA</h1>
```

```
<?php
```

```
mysql_connect("localhost", "pradera", "pradera") or die ("Could not connect: " .
```

```
mysql_error());
```

```
mysql_select_db("Ganado");
```

```
$result = mysql_query("SELECT * FROM Vaca1");
```

```
echo "<table border='1'>
```

```
<tr>
```

```
<th>ID</th>
```

```
<th>Medicacion</th>
```

```
<th>Fecha</th>
```

```
<th>Tiempo</th>
```

```
<th>Ordeno</th>
```

```
<th>Numero</th>
```

```
</tr>";
```

```
/*<input type="button" value="Inicio" title="Inicio" header('Location:
```

```
http://192.168.1.15/pradera/index.php')/>*/
```

```
while($row = mysql_fetch_array($result)){
```

```
    echo"<tr>";
```

```
    echo"<td>" . $row['Id'] . "</td>";
```

```
    echo"<td>" . $row['Medicacion'] . "</td>";
```



```

echo"<td>" . $row['Fecha'] . "</td>";

echo"<td>" . $row['Tiempo'] . "</td>";

echo"<td>" . $row['Ordeno'] . "</td>";

echo"<td>" . $row['Numero'] . "</td>";

echo"</tr>";

}

echo "</table>";

mysql_close($con);

?>

<br />

<input type="button" value="Inicio" title="Inicio"
onClick="location.href='http://192.168.1.15/pradera/index.php'"/>

</body>

</div>

<img src='footvacas.jpg' width=1400 height=50 vspace=3 hspace=0/>

</html>

```

Script de la Consulta Vaca2 en la base de datos: vaca2.php

```

<html>
<head>
<img src='cbvaca.jpg' width=1400 height=350 vspace=3 hspace=0/>

```

```

<div align=center>
</head>
<title>vaca2</title>
<body>
<h1>DESCRIPCION DE ISABELLA</h1>

<?php
mysql_connect("localhost", "pradera", "pradera") or die ("Could not connect: " .
mysql_error());
mysql_select_db("Ganado");

$result = mysql_query("SELECT * FROM Vaca2");

echo "<table border='1'>
<tr>
<th>Sample Number</th>
<th>Medicion</th>
<th>Fecha</th>
<th>Tiempo</th>
<th>Ordeno</th>
<th>Numero</th>
</tr>";

while($row = mysql_fetch_array($result)){
    echo"<tr>";
    echo"<td>" . $row['Id'] . "</td>";
    echo"<td>" . $row['Medicion'] . "</td>";
    echo"<td>" . $row['Fecha'] . "</td>";
    echo"<td>" . $row['Tiempo'] . "</td>";
    echo"<td>" . $row['Ordeno'] . "</td>";
    echo"<td>" . $row['Numero'] . "</td>";
    echo"</tr>";
}

```

```
echo "</table>";  
mysql_close($con);  
?>
```

```
<br />  
<input type="button" value="Inicio" title="Inicio"  
onClick="location.href='http://192.168.1.15/pradera/index.php'"/>  
</body>  
</div>  
<img src='footvacas.jpg' width=1400 height=50 vspace=3 hspace=0/>  
</html>
```

ANEXO 04: CÁLCULOS DE LOS RESULTADOS OBTENIDOS DEL SENSOR DE FLUJO PARA VER LA CANTIDAD DE LECHE PRODUCIDA EN LITROS DE UN BOVINO.

1. El sensor de flujo envía un número de pulsaciones al microcontrolador, el cual el microcontrolador es el encargado de recibir estas pulsaciones a través del recurso de la interrupción.

$$\text{Contapulso} = \text{Numero de pulsaciones.}$$

2. El microcontrolador calcula el flujo de caudal a través de la siguiente formula:

$$\text{Caudal} = (\text{Contapulso} / x)$$

Donde x es la variable que se debe calcular empíricamente para obtener el flujo del caudal en Litros/min.

3. El microcontrolador realiza una sumatoria del resultado de la variable Caudal cada segundo. Donde la variable media se inicializa en 0.

$$\text{Sumatoria} = \text{Sumatoria} + \text{Caudal}$$

4. Cuando el sensor de flujo ya no tenga nada que medir, el microcontrolador realiza un promedio de la sumatoria en el tiempo transcurrido. Ese tiempo también es obtenido por el microcontrolador y es en segundos.

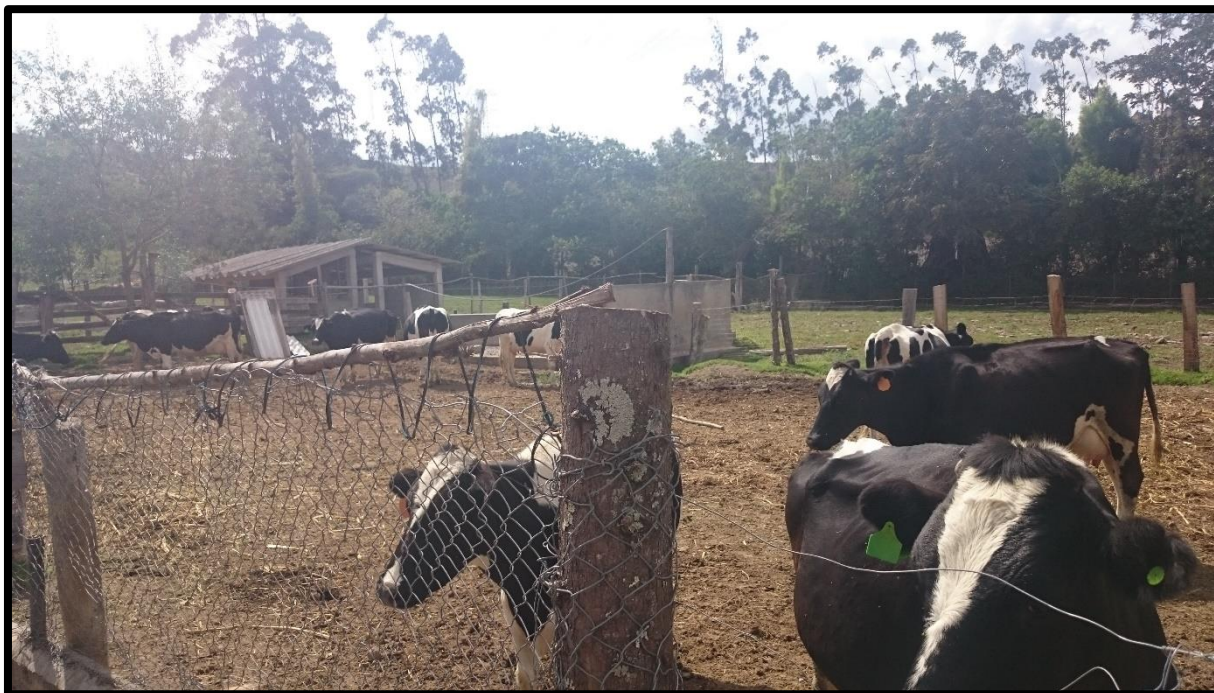
$$\text{Promedio} = \text{Sumatoria/tiempo}$$

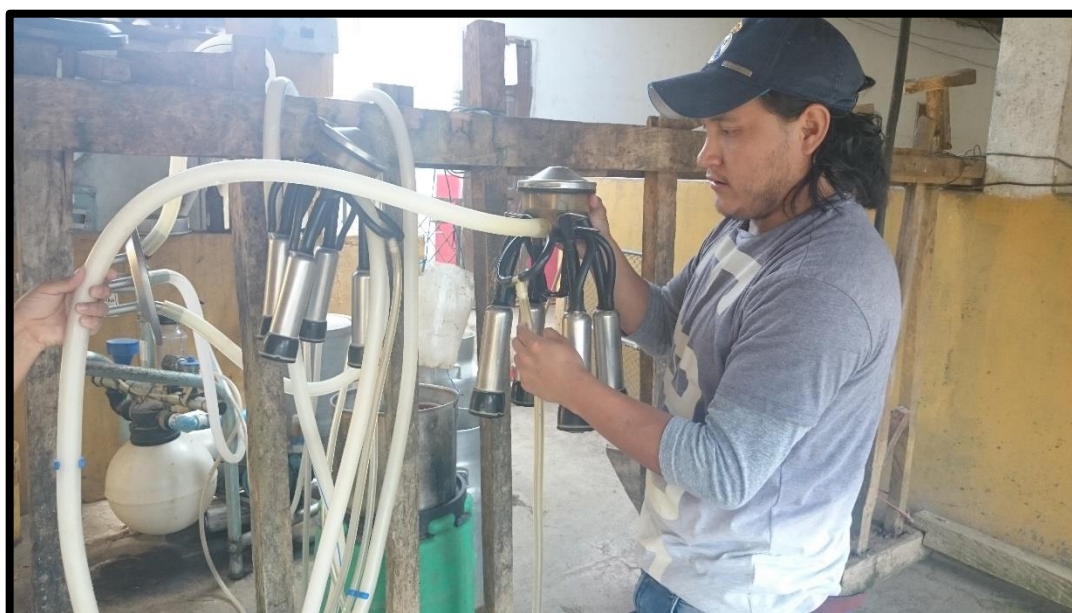
5. Finalmente para obtener la cantidad de leche producida del bovino, el microcontrolador ejecuta la siguiente operación:

$$\text{Total} = (\text{Promedio} * \text{tiempo}) / 60$$

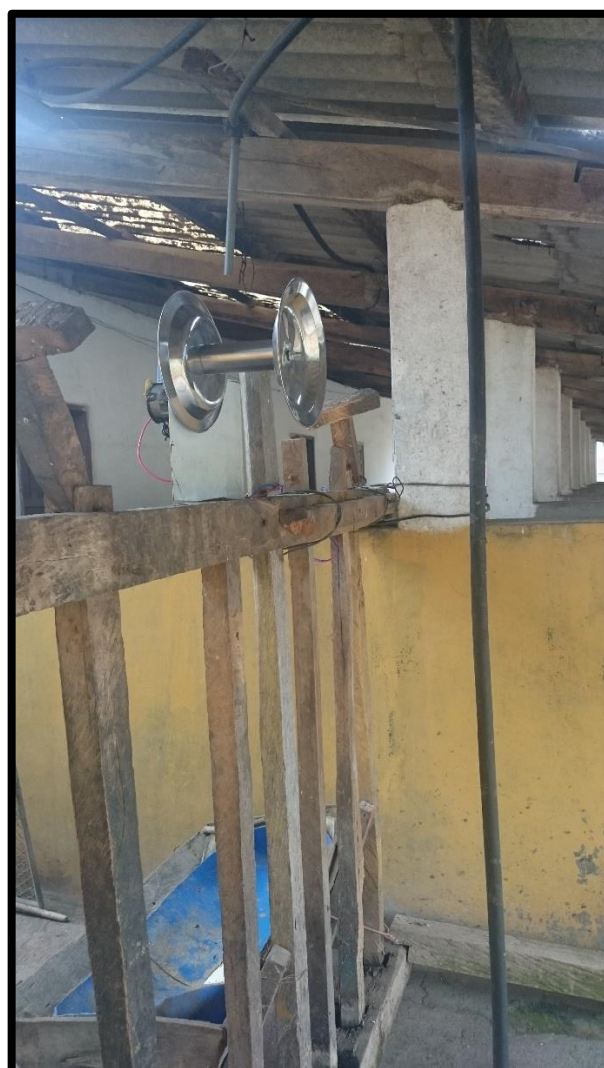
El Total es el Litros.

ANEXO 05: IMÁGENES











MANUALES

(Los manuales tanto de administrador como de operador se encuentran en formato digital en el CD)