

**SISTEMA DE CONTROL AUTOMÁTICO PROGRAMABLE
PARA EL RIEGO POR GOTEO
DE UN HUERTO URBANO EN MACETAS**

Lic. Gustavo Eloy Alcaráz Bogado

Tutor: Ing. Julio César Morínigo Bauza

**Tesis presentada en la Universidad Tecnológica Intercontinental (UTIC) como requisito
parcial para la obtención del título de Ingeniero en Sistemas Informáticos**

Asunción, 2019

Derecho de Autor

Quien suscribe, Gustavo Eloy Alcaráz Bogado, con Documento de Identidad No 2.351.696, autor del trabajo de investigación titulado "**SISTEMA DE CONTROL AUTOMÁTICO PROGRAMABLE PARA EL RIEGO POR GOTEO DE UN HUERTO URBANO EN MACETAS**" otorga a la UTIC la facultad de comunicar la obra, divulgar, publicarla y reproducirla en soportes analógicos o digitales en la oportunidad que ella así lo estime conveniente. La UTIC debe indicar que la autoría o creación del trabajo corresponde a mi persona y hará referencia al tutor y a las personas que hayan colaborado en la realización del presente trabajo de investigación.

En la ciudad de Asunción, a los días 3 del mes de diciembre de 2.019.

Gustavo Eloy Alcaráz Bogado

C.I. 2.351.696

Constancia de Aprobación del Tutor

Quien suscribe, Ing. Julio César Morínigo Bauzá, con documento de Identidad N° 1.047.978, tutor del trabajo de investigación titulado **«SISTEMA DE CONTROL AUTOMÁTICO PROGRAMABLE PARA EL RIEGO POR GOTEO DE UN HUERTO URBANO EN MACETAS»**, elaborado por el alumno, Gustavo Eloy Alcaráz Bogado, para obtener el Título de Ingeniero en Sistemas Informáticos hace constar que dicho trabajo reúne los requisitos exigidos por la Facultad de Tecnología Informática de la Universidad Tecnológica Intercontinental y puede ser sometido a evaluación y presentarse ante los docentes que fueron designados para integrar la Mesa Examinadora.

En la ciudad de Asunción, a los 3 días del mes de diciembre de 2.019.

Dedicatoria

Dedicado a

Agradecimientos

TABLA DE CONTENIDO

Derecho de Autor	I
Constancia de Aprobación del Tutor.....	II
Dedicatoria.....	III
Agradecimientos	IV
TABLA DE CONTENIDO	XIV
LISTA DE FIGURAS.....	XVII
LISTA DE TABLAS	XIX
Portada	1
Resumen	2
Summary.....	3
Diagnóstico	4
Resultado del diagnóstico	4
Relevamiento	4
Técnica de la observación	4
Ambiente físico.....	4
Ambiente social y humano.....	5
Técnica de la entrevista estructurada.....	6
Preguntas teniendo en cuenta las macetas.....	6
Preguntas teniendo en cuenta el riego por goteo.....	6
Planteamiento del Problema.....	8
Pregunta General.....	9
Preguntas Específicas	9
Sustento Teórico	10
Antecedentes	10
Nelson modelo 56607.....	10
Gardena modelo MasterControl.....	10
K-rain modelo ProEX 2.0 Wi-Fi Enabled Controller.....	11
Bases teóricas	12
Huerto urbano.....	12
Beneficios.....	13
Cultivo en macetas.....	13
El sustrato.....	14
El contenedor adecuado.....	14

Riego por goteo o riegos localizados a baja presión.....	15
Componentes del riego localizado por goteo.....	15
Algunos conceptos básicos de hidráulica.....	16
Diseño de una instalación de riego.....	16
Fundamentos de las ciencias de ingeniería.....	17
Sistemas de control.....	18
Manual.....	18
Automático.....	18
Elementos de un sistema de control automático.	18
Tipos de sistemas de control.....	20
Lazo abierto.....	20
Lazo cerrado.	21
Clasificación de los sistemas de control.	22
Dimensión.	22
Conocimiento de sus parámetros.	22
Carácter de transmisión en el tiempo.	22
Presencia de linealidad.	23
Comportamiento en el tiempo.	23
Características del control.	23
Métodos de control.....	24
Clásico.	25
Moderno.	25
Avanzado.....	26
Estrategias de control.	28
Cascada (Cascade).....	28
Relación (Ratio).	28
Rango dividido (Split range).	29
Selectivo (Override).	29
Inferencial.....	29
Compensación de tiempo muerto.	29
Mapa conceptual de sistema de control.	30
Sistemas de automatización.....	30
Autómatas programables.	31
Funciones del autómata y del sistema.	31
Estructura de un autómata.....	32
Selección del autómata.	32
Ciclos del programa.	34
Fundamentos de la electricidad.	35
Corriente continua y corriente alterna.....	35

Resistencia eléctrica y potencia eléctrica	35
Conductor eléctrico.....	35
Resistencia eléctrica.....	36
Ley de Ohm.....	36
Potencia eléctrica.	36
Funcionamiento de un electroimán.....	36
Estructura de un electroimán.....	37
Aplicaciones de electroimanes.....	37
Funcionamiento de un condensador eléctrico.	37
Funcionamiento de un diodo.	38
Funcionamiento y estructura de interruptores y conmutadores....	38
Relés y contactores.....	39
Aplicaciones de relés.	39
Estructura de un relé.....	39
Funcionamiento y construcción de una unidad de alimentación...	40
Mediciones en un circuito eléctrico.	40
Fundamentos de la electrónica.....	42
Clasificación de los componentes electrónicos.....	43
Componentes pasivos.	43
Componentes activos.	43
Resistencias.	44
Tolerancia de una resistencia.....	44
Código de colores para resistencias.....	45
Potencia de disipación de una resistencia.	45
Clasificación de las resistencias.	46
Condensadores.	47
Aplicaciones de los condensadores.	47
Tipos de condensadores.	48
Condensadores variables.	48
Codificación de los condensadores.	49
Bobinas.	49
Tipos de bobinas.	49
Conductores eléctricos.	51
Clasificación de los conductores eléctricos.	51
Interruptores.	52
Conectores.....	52
Fusibles.	53
Diodos.	53
Maneras de conectarse un diodo.	53

Tipos de diodos.....	53
Transistores.....	55
Circuitos integrados.....	55
Sistema electrónico.....	57
Microcontrolador.....	57
Señales de un sistema electrónico.....	58
Protoboard.....	58
Fundamentos de la informática.....	59
Estructuras de un sistema de computación.....	59
Hardware.....	59
Placa Base.....	60
Unidad Central de Proceso (CPU).....	61
Memoria: Funciones, tipos y características.....	62
Dispositivos de Entrada/Salida.....	64
Software.....	65
Clasificación del software.....	65
Lenguajes de Programación.....	67
Lenguaje C.....	67
Python.....	68
Php.....	69
JavaScript.....	70
Entorno de desarrollo integrado (IDE).....	70
Arduino (IDE).....	70
Sistemas Operativos.....	71
Sistema Operativo OpenWrt	72
Base de Datos.....	72
Definición.....	72
Sistema de gestión de bases de datos.....	72
Lenguaje SQL.....	73
Sqlite.....	74
Redes.....	76
Definiciones.....	76
Tipos de redes.....	77
Topología de red.....	78
Medios físicos de conexión de red.....	78
Topología física de redes.....	78
Métodos de acceso al soporte.....	80
Componentes de una red.....	81
IEEE 802.....	82

IEEE 802.1 (gestión y interconexión de redes)	82
IEEE 802.2 (LLC).....	82
IEEE 802.3 y Ethernet.....	83
IEEE 802.5 y Token Ring.....	83
IEEE 802.11 y Wi-Fi.....	84
IEEE 802.15 y Bluetooth.....	87
Arquitectura de protocolos de red.....	88
Internet.....	90
Direcciones IP.....	90
Máscara de red.....	91
DHCP (Dinamyc Host Configuracion Protocol).....	91
SSH (Secure SHell).....	91
FTP (File Transfer Protocol).....	91
DNS (Domain Name Systems).....	92
HTTP (HyperText Transfer Protocol).....	92
SMTP (Simple Mail Transfer Protocol).....	94
POP3 (Post Office Protocol).....	94
IMAP (Internet Message Access Protocol).....	94
Transmisión de vídeo en tiempo real (video streaming).....	94
Definiciones.....	94
Video.....	94
Formato de video (video format).....	95
Códec (compresor).....	95
Cuadro (frame).....	95
Tamaño del cuadro (frame size).....	95
Relación de aspecto del cuadro (frame aspect ratio).....	96
Cuadros por segundo (frame rate).....	96
Método de exploración (scanning method).....	96
Arquitectura de protocolos de un sistema de video streaming.....	97
RSTP.....	97
RTP.....	97
RTCP.....	97
HTTP.....	98
Herramientas de transmisión de video sobre HTTP.....	99
Motion.....	99
Mjpeg-streamer.....	99
Cámara web.....	100
Definición.....	100
Características.....	100

Software.....	100
Tecnología.	101
Sistemas embebidos o empotrables.....	101
Definición.	101
Función.	101
Arquitectura.	101
Lenguaje de programación.	102
El proceso de desarrollo.	102
Fases de diseño.	103
Placas bases, módulos, periféricos y fuentes de alimentación.	104
Placas bases embebidas para la unidad de control.....	104
Arduino (placa de desarrollo).	104
Raspberry Pi (micro computador).	107
Beagleboard.org (placa de desarrollo).	108
Comparativas de placas bases.....	110
Módulos pre-actuadores para accionar al actuador.	112
Relé electromecánico.....	112
Relé de estado sólido (SSR).....	112
Comparativos de módulos relé de un canal.	112
Módulos actuadores para accionar el riego.	113
Electroválvula de corriente alterna.	113
Electroválvula de corriente continua.	113
Comparativos de electroválvulas para agua.	113
Módulos sensores para temperatura y humedad relativas.....	114
DHT11.	114
DHT21.	114
DHT22.	115
Comparativos de sensores de temperatura y humedad relativa.....	115
Módulos sensores para humedad del suelo.	116
Adafruit Stemma.	116
YL-69/YL-38.	116
KS0049.	117
Comparativos de sensores de humedad de suelo.	117
Módulos sensores para detección de lluvia.	118
MH-RD / YL-38.....	118
Octopus EF04024.	118
Keyes T1592.....	118
Comparativos de sensores de detección de lluvia.	118

Módulos sensores para detección de luz	119
BH1750.	119
CdS LDR GL5528.....	119
MOD-LDR / YL-38.	119
Comparativos de sensores para detección de luz.....	120
Módulos sensores para medir flujo del líquido (caudalímetro). ...	120
FS300A.	120
YF-S201.	121
FS400A.	121
Comparativos de sensores para medir flujo del líquido	121
Módulos de reloj de tiempo real (RTC).	122
RTC DS1307.....	122
DS3231MPMB1.....	122
BOB-14558.	123
Comparativo de módulos de reloj de tiempo real (RTC).	123
Periféricos de indicadores luminosos (salida).	124
LED-01.....	124
LED-RGB-B5C.....	124
LED-MATRIZ-MAX7219.	124
Comparativos de indicadores luminosos.	125
Periféricos de tarjetas de memorias de almacenamiento de datos	
(entrada/salida).	125
Comparativo de tarjetas de memorias de almacenamiento	
de datos.	126
Periféricos de video captura web (entrada).	127
Comparativos de cámaras web.....	127
Fuentes de alimentación de energía.	127
Comparativos de fuentes de alimentación de energía.....	128
Aspectos Legales.....	129
Ley N° 3239. De los Recursos Hídricos del Paraguay. República del	
Paraguay (2007)	129
Capítulo I: Objetivo.....	129
Capítulo II: Principios	129
Capítulo III: Definiciones	130
Capítulo VI: Derechos de Uso y Aprovechamiento de los	
Recursos Hídricos.....	130
Ley N° 716. Que sanciona delitos contra el medio ambiente.	
(República del Paraguay, 1996).....	131
Artículo 5º.-	131

Artículo 14º-.....	131
El Plan Nacional de Desarrollo Paraguay 2030 (PND 2030)	131
D. Marco estratégico.....	131
Líneas transversales	131
Proyecto de Intervención.....	132
Objetivo General	132
Objetivos Específicos	132
Justificación	132
La descripción del producto o situación esperada	132
Las proporciones de cliente y/o mercado potencial	133
La proyección financiera que se espera con la implementación del proyecto	134
Los involucrados del proyecto, con sus respectivas responsabilidades	134
Descripción de las Etapas	135
Etapa 1: Inicial.	135
Exploración de bases teóricas.....	135
Búsqueda de referencias técnicas.	135
Elección y obtención del hardware para el dispositivo.	135
Etapa 2: Hardware del dispositivo.	135
Elaboración del esquema eléctrico entre módulos.	135
Creación de placa de circuito impreso de conexión entre módulos.	135
Ensamblado de los módulos.	135
Etapa 3: Software del dispositivo.	136
Elaboración del flujo de control automatizado.....	136
Instalación y configuración de aplicativos para la placa base embebida.....	136
Creación de una estructura de almacenamiento de datos....	136
Programación del firmware de control automatizado.....	136
Creación del programa externo para el servicio de telemetría.	136
Creación del programa externo para el servicio de notificación de alertas.....	136
Creación del programa externo de estrategia de control de riego.	136
Creación del programa externo de activación de la cámara web.	137
Creación del software de interfaz gráfica del usuario.....	137
Etapa 4: Final.....	137
Adecuación del hardware del dispositivo en cajas plásticas externas.....	137

Pruebas y ajustes finales	137
Redacción del informe final.....	137
Diagrama de Gantt del desarrollo de las etapas	138
Plan de Trabajo	139
Desarrollo de las Etapas	139
Etapa 1: Inicial.	139
Exploración de bases teóricas	139
Búsqueda de referencias técnicas.....	142
Elección y obtención del hardware para el dispositivo.	142
Etapa 2: Hardware del dispositivo.	145
Elaboración del esquema eléctrico entre módulos.	145
Creación de placa de circuito impreso de conexión entre módulos.	148
Ensamblado de los módulos.	149
Etapa 3: Software del dispositivo.	151
Elaboración del flujo de control automatizado.....	151
Instalación y configuración de aplicativos para la placa base embebida.....	157
Preparación de una estructura de almacenamiento de datos..	163
Programación del firmware de control automatizado.....	167
Creación del programa externo para el servicio de telemetría.	168
Creación del programa externo para el servicio de notificación de alertas.....	170
Creación del programa externo de estrategia de control del riesgo.	171
Creación del programa externo de activación de la cámara web.	173
Creación del software de la interfaz gráfica del usuario.....	174
Etapa 4: Final.....	183
Adecuación del hardware del dispositivo en cajas plásticas externas.....	183
Pruebas y ajustes finales.	185
Redacción del informe final.....	189
Recursos utilizados por Etapa	191
Diagrama de Gantt Ejecutado	192
Metodología.....	193
Tipo de Investigación	193
Modalidad	193
Conclusiones y Recomendaciones	194

REFERENCIAS	200
GLOSARIO DE TÉRMINOS	203
Anexo A	204
Anexo B	208
Anexo C	213
Anexo D	224

LISTA DE FIGURAS

1	Programador de riego 56607 Nelson. Fuente: Nelson (2011)	10
2	Programador de riego MasterControl Gardena. Fuente: Gardena (2013)	11
3	Programador de riego ProEX 2.0 Wi-Fi Enabled Controller K-Rain. Fuente: K-Rain (2015).....	11
4	La caja negra de un sistema técnico. Fuente: Ebel <i>et al.</i> (2007).....	17
5	Ejemplo control manual. Fuente: (Estefania, 2017).....	18
6	Diagrama en bloques de los elementos del sistema de control automático. Fuente: Mendiburu (2003).....	19
7	Diagrama en bloques del sistema de control en lazo abierto. Fuente: Mendiburu (2003).....	21
8	Diagrama en bloques del sistema de control en lazo cerrado. Fuente: Mendiburu (2003).....	22
9	Mapa conceptual sobre sistema de control. Fuente: Autor	30
10	Corriente continua y corriente alterna en función del tiempo. Fuente: Ebel <i>et al.</i> (2007).....	35
11	Bobina eléctrica sin y con núcleo de hierro y las correspondientes líneas de campo magnético. Fuente: Ebel <i>et al.</i> (2007).....	37
12	Funcionamiento de un condensador. Fuente: Ebel <i>et al.</i> (2007).....	38
13	Funcionamiento de un diodo. Fuente: Ebel <i>et al.</i> (2007)	38
14	Contacto normalmente abierto, contacto normalmente cerrado y conmutador: vista en corte y símbolo respectivamente. Fuente: Ebel <i>et al.</i> (2007)	39
15	Relé: vista en corte y símbolo. Fuente:Ebel <i>et al.</i> (2007)	40
16	Grupos de la unidad de un sistema de control electroneumático. Fuente: Ebel <i>et al.</i> (2007)	40
17	Aparato de medición universal. Fuente: Ebel <i>et al.</i> (2007)	41
18	Medición de tensión y de intensidad respectivamente. Fuente: Ebel <i>et al.</i> (2007)	42
19	Medición de resistencia. Fuente: Ebel <i>et al.</i> (2007)	42
20	Sistema electrónico. Fuente: Artero (2013).....	57
21	Protoboard. Fuente: Plastica (2018).....	59
22	Logo Wi-Fi. Fuente: Dordogne (2015).....	84
23	Interfaz Host-Target para programación del sistema embebido. Fuente: Arriarán (2015).....	102
24	Diagrama de fases para el diseño de un sistema empotrado. Fuente: Guillén (2011)104	

25	Arduino Uno. Fuente: Arduino (2019).....	106
26	Arduino Yún. Fuente: Arduino (2019).....	106
27	Arduino Tian. Fuente: Arduino (2019)	107
28	RaspBerry Pi 3 Model B. Fuente: Pi (2019).....	108
29	Beaglebone Black Wireless. Fuente: BeagleBoard.org (2018)	109
30	Pre-Actuadores Songle, Tongling y Omron respectivamente. Fuente:Iberorobotic (2019), Embajadores (2018)	112
31	Electroválvulas para agua AC-DC respectivamente. Fuente: Mechatronics (2019)	113
32	Módulos de sensores de temperatura y humedad relativa. Fuente: Mechatronics (2019).....	115
33	Sensores de humedad del suelo. Fuente: Didactic (2019); Electronics (2019); Mechatronics (2019).....	117
34	Sensores de detección de lluvia. Fuente: Caldas (2013); Electrónica (2014) ..	118
35	Sensores para detección de luz. Fuente: Mechatronics (2019)	120
36	Sensores para medir flujo del líquido. Fuente: Mechatronics (2019)	121
37	Módulos de reloj de tiempo real (RTC). Fuente: Electronics (2019); Mechatronics (2019).....	123
38	Indicadores luminosos. Fuente: Electronics (2019)	125
39	Tarjetas de memorias de almacenamiento de datos. Fuente: Electronics (2019) ..	126
40	Cámaras web. Fuente: Logitech (2019); Microsoft (2019); Razer (2019)	127
41	Fuentes de alimentación de energía. Fuente: Electronics (2019); Mechatronics (2019).....	128
42	Montaje de los módulos del dispositivo en protoboard. Fuente: Autor.....	146
43	Diagrama electrónico o esquema eléctrico del dispositivo. Fuente: Autor	147
44	Herramienta para proyectos de sistemas electrónicos EasyEDA. Fuente: Autor..	147
45	Pistas del PCB del dispositivo. Fuente: Autor	148
46	Proceso del PCB para el dispositivo. Fuente: Autor.....	149
47	Proceso del ensamblado de los módulos. Fuente: Autor	150
48	Diagrama de flujo principal del Firmware para el dispositivo. Fuente: Autor.....	151
49	Diagrama de flujo del programa externo para servicio de datos de sensores (datos_sensores). Fuente: Autor.	153
50	Diagrama de flujo del programa externo para el servicio de notificación de alertas (notificaciones_alertas). Fuente: Autor.	154
51	Diagrama de flujo del programa externo para la estrategia de control del riego (estrategia_control). Fuente: Autor.	155
52	Diagrama de flujo del programa externo para utilizar la cámara web (camara_web). Fuente: Autor.	156
53	Herramienta de diagramación draw.io. Fuente: Autor.	156
54	Configuración del Arduino Yún. Fuente: Autor	157

55	IDE de desarrollo Arduino Yún. Fuente: Autor	159
56	WinSCP. Fuente: Autor	160
57	PuTTY. Fuente: Autor	162
58	Visual Studio Code. Fuente: Autor	163
59	Diagrama entidad relación para el dispositivo. Fuente: Autor.	165
60	Herramienta administrador de base de datos SQLiteStudio. Fuente: Autor.	166
61	Bloc de notas con Script SQL de muestra de la tabla especies. Fuente: Autor.....	166
62	IDE de desarrollo Arduino con el sketch del firmware. Fuente: Autor	168
63	Puente entre Microcontrolador ATmega32u4 y Linux OpenWrt del Arduino Yún. Fuente: Autor.....	169
64	datos_sensores.py (código fuente). Fuente: Autor.....	170
65	notificaciones_alertas.py (código fuente). Fuente: Autor	171
66	estrategia_control.py (código fuente). Fuente: Autor	173
67	camara_web.py (código fuente). Fuente: Autor	174
68	Captura pantalla: presentación, control de acceso, página principal y menú activado. Fuente: Autor	175
69	Captura pantalla: informaciones. Fuente: Autor.....	176
70	Captura pantalla: especies. Fuente: Autor	176
71	Captura pantalla: configuraciones. Fuente: Autor	177
72	Captura pantalla: supervisión. Fuente: Autor	177
73	Captura pantalla: riegos. Fuente: Autor	178
74	Captura pantalla: minería de datos. Fuente: Autor	178
75	Captura pantalla: tarifas de agua. Fuente: Autor	179
76	Captura pantalla: históricos. Fuente: Autor	179
77	Captura pantalla: usuarios. Fuente: Autor	180
78	Captura pantalla: fecha/hora dispositivo y perfil. Fuente: Autor	180
79	Código fuentes de la vista, el modelo y el control para usuarios. Fuente: Autor ..	181
80	Esquema de funcionamiento del dispositivo. Fuente: Autor	182
81	Cajas plásticas externas terminadas para el dispositivo. Fuente: Autor	184
82	Ambiente de prueba. Fuente: Autor	187
83	Datos obtenidos 02/12/2019. Fuente: Autor	188
84	LyX. Fuente: Autor	189
85	JabRef. Fuente: Autor.....	190

LISTA DE TABLAS

1	Recipientes adecuados según cultivo. Fuente: Schonwald y Pescio (2015).....	14
2	Código de colores para resistencias. Fuente: Miguel (2014)	45
3	Ejemplos de resistencias SMD con tres dígitos. Fuente: Miguel (2014)	45
4	Tipos resistencias fijas. Fuente: Miguel (2014)	46
5	Tipos condensadores fijos. Fuente: Miguel (2014).....	48
6	Código de colores para condensadores. Fuente: Miguel (2014)	49
7	Clasificación de circuitos integrados. Fuente: Schilling y Charles (1991)	56
8	Clasificación de los procesadores interconectados con base en la escala. Fuente: Tanenbaum y Wetherall (2012).....	78
9	Las normas 802.x del IEEE. Fuente: Dordogne (2015)	82
10	Arquitectura de la pila de protocolos TCP/IP. Fuente: Olifer y Olifer (2009)	90
11	Clases de direcciones IP. Fuente: Olifer y Olifer (2009)	90
12	Comparativo Microcontrolador. Fuente: Autor	110
13	Comparativo Microprocesador. Fuente: Autor	110
14	Comparativo Interfaces. Fuente: Autor	111
15	Comparativo General. Fuente: Autor	111
16	Comparativo de módulos de relé de un canal. Fuente: Autor	113
17	Comparativos de electroválvulas para agua AC-DC. Fuente: Autor	114
18	Comparativos de sensores de temperatura y humedad relativa. Fuente: Autor	116
19	Comparativo de sensores humedad del suelo. Fuente: Autor	117
20	Comparativo de sensores de detección de lluvia. Fuente: Autor.....	119
21	Comparativos de sensores para detección de luz. Fuente: Autor	120
22	Comparativos de sensores para medir flujo del líquido. Fuente: Autor	122
23	Comparativos de módulos de reloj de tiempo real (RTC). Fuente: Autor	124
24	Comparativo indicadores luminosos. Fuente: Autor	125
25	Comparativo tarjetas de memorias de almacenamiento de datos. Fuente: Autor ..	126
26	Comparativo de cámaras web. Fuente: Autor	127
27	Comparativo de fuentes de alimentación de energía. Fuente: Autor	128
28	Proyección financiera. Fuente: Autor	134
29	Disposición de conexión de pines del Arduino Yún. Fuente: Autor	145
30	Lista de paquetes instalados en el S. O. Linux OpenWrt del Arduino Yún mediante el aplicativo PuTTY. Fuente: Autor	161
31	Datos cargados en la tabla historicos_motivos. Fuente: Autor.....	167

32	Comparativo utilizado por la Estrategia de Control del riego en el día. Fuente: Autor.....	172
33	Disposición del hardware en cajas plásticas externas. Fuente: Autor	183
34	Recursos utilizados en las Etapas. Fuente: Autor.....	191

Sistema de Control Automático Programable
para el Riego por Goteo
de un Huerto Urbano en Macetas

Lic. Gustavo Eloy Alcaráz Bogado
Universidad Tecnológica Intercontinental

Carrera de Ingeniería en Sistemas Informáticos, Sede 1
geab1073@gmail.com

Resumen

Este trabajo pretende solucionar el inconveniente de las personas que desean implementar un área ecológica en su hábitat, mediante macetas con riego por goteo sin mecanismo de automatización en el proceso, y que muchas veces encuentran inconvenientes para sostener un mantenimiento constante, que por cuestiones de tiempo, actividades varias que desarrollan una persona en la actualidad, así como de subsanar el margen de error humano en la técnica del riego y optimizar el uso del agua a través de una adaptativa tecnológica para un control automático programable, basados en los indicadores ambientales adecuados a la especie cultivada.

El resultado del trabajo podría considerarse como un marco de referencia para futuros proyectos de implementaciones tecnológicas, en las áreas de la agricultura y la botánica, considerando los métodos de recolección y almacenamiento de datos tomados directamente del ambiente en línea y de manera precisa para su posterior análisis por personas idóneas y especializadas en las ramas mencionadas, con la finalidad de perfeccionar la toma de decisiones.

En cuanto a las áreas de tecnología de información y comunicación aporta variados beneficios como interactuar el hardware y el software con el medio ambiente sin la intervención humana, tomando información de la naturaleza y registrando la misma en una base de datos, además de apuntar hacia el uso de herramientas de código abierto como así también de componentes electrónicos de bajos costos; y por último incorporar este trabajo como una guía para futuras soluciones o aplicaciones tecnológicas específicas a desarrollar.

Palabras clave: agua, riego, goteo, cultivo, huerto urbano, medio ambiente, zona verde, tecnología verde, sensores ambientales, sistema embebido, sistemas de control, automatización, telemetría.

Summary

This work aims to solve the inconvenience of people who wish to implement an ecological area in their habitat, by means of pots with drip irrigation without automation mechanism in the process, and that many times find inconveniences to sustain a constant maintenance, which for reasons of time , various activities that a person is currently developing, as well as correcting the margin of human error in the irrigation technique and optimizing the use of water through a technological adaptive for a programmable automatic control, based on the appropriate environmental indicators to the cultivated species.

The result of the work could be considered as a benchmark for future projects of technological implementations, in the areas of agriculture and botany, considering the methods of data collection and storage taken directly from the online environment and precisely for later analysis by suitable and specialized people in the aforementioned branches, in order to improve decision making.

As for the areas of information and communication technology, it offers several benefits such as interacting hardware and software with the environment without human intervention, taking information from nature and registering it in a database, in addition to pointing towards the use of open source tools as well as low cost electronic components; and finally incorporate this work as a guide for future solutions or specific technological applications to develop.

Keywords: water, irrigation, drip, cultivation, urban garden, environment, green area, green technology, environmental sensors, embedded system, control systems, automation, telemetry.

Diagnóstico

Resultado del diagnóstico

Relevamiento. Se procedió a realizar la investigación empleando técnicas de recopilación de información con el fin de validar y obtener la confiabilidad en el diagnóstico, en primer lugar fue empleado la técnica de la observación tanto del ambiente físico, social y humano, y en segundo lugar se empleó la técnica de la entrevista estructurada para obtener datos técnicos puntuales para el proyecto de automatización programable del riego por goteo de un huerto urbano en macetas.

A continuación se expone lo recabado de acuerdo a cada técnica empleada.

Técnica de la observación. Mediante la técnica de la observación aplicada durante una semana, según disponibilidad de tiempo realizado en fecha del 10 al 16 de mayo de 2019, se presenta en forma agrupada según las cuestiones que son importantes de acuerdo a lo observado (Hernández *et al.*, 2010, p.412):

Ambiente físico. Huerto urbano con un área de 3,685 metros cuadrados de espacio utilizado, cuyas dimensiones son de 1 metro de ancho por 2,5 metros de largo ubicado al costado izquierdo de la edificación en el patio lateral de mi residencia en sitio calle de la Amistad Nro. 172 de la ciudad de San Lorenzo del Departamento Central de la República del Paraguay, se compone por una distribución de 10 macetas jardineras exteriores en dos filas y separadas 40 cm entre sí sobre un suelo revestido por pisos de tipo cerámico poroso con diseño de piedras estilo rompecabezas, las macetas jardineras están fabricadas en material plástico de color marrón oscuro en forma de cono recto truncado invertido teniendo 4 agujeros pequeños en el fondo para el drenaje, sus dimensiones son de 30 centímetros de diámetro de base superior, 30 centímetros de apotema (altura) y 20 centímetros diámetro en la base inferior, cuentan con sus respectivos platos de retención de líquido excedente con un diámetro de 25 centímetros, la maceta cuenta con un volumen de capacidad de aproximadamente 14,92 litros, conteniendo en su interior un sustrato compuesto por una mezcla de humus de lombriz con tierra normal y como base piedra tritura número 4, en ellas se encuentran cultivadas plantas de tomate cherry (*Solanum lycopersicum* variedad Cerasiforme) teniendo una altura promedio de 38 centímetros, el huerto urbano tiene implementado un proceso manual de riego por goteo, compuesto por 15 metros de micro tubo de 6 milímetros de diámetro exterior y 4 milímetros de diámetro interior, 10 goteros regulables con piqueta con una capacidad de suministro de agua de 6 litros/horas, 11 uniones tipo te (T), 2 codos de 90°, 2 terminadores de línea y 1 conector con reducción de boca de 3/4 pulgadas a 4 milímetros conectada en forma directa a la red de distribución de agua corriente local (ESSAP) por medio de una canilla para lavarropas con un pico de 3/4 de pulgadas, por último el área investigada está cubierta con un techo translucido en policarbonato con una altura de 2,40 metros que deja pasar en gran medida la luz solar y se encuentra separada a 1 metro de la pared de la

casa teniendo un corredor de aire para su ventilación hacia la calle. (ver Anexo A)

Ambiente social y humano. Están compuesta por una persona de la tercera edad en situación de dependencia directa, dos personas adultas con estudios universitarios con horarios de trabajo entre semana de lunes a viernes de 07:00 hs a 15:00 hs, más un tiempo de retorno del trabajo de casi dos horas y un menor de edad en etapa escolar con un horario de 07:00 hs a 11:20 hs de lunes a viernes, el ambiente observado es manipulado según necesidad y tiempo disponible por los integrantes, en especial los fines de semana.

Actividades (acciones) individuales y colectivas: El huerto urbano es mantenido prácticamente por la persona de la tercera edad, teniendo como acciones; examinar las plantas, abrir/cerrar la llave de la canilla para el riego según necesidad y/o controlar las condiciones físicas de las macetas sin tener un horario específico en el día o frecuencias determinadas para las tareas y casi siempre según el tiempo libre, exceptuando los días feriados o fines de semana donde participan la mayoría de los integrantes como distracción y entretenimiento, se evidencia que al momento del riego no se tienen en cuenta horarios, duraciones ni condiciones ambientales factibles en los días de la semana para la apertura/cierre de la canilla de suministro de agua, más bien por criterios individuales o estados de ánimo de los involucrados, es decir, en ocasiones es escasa, nula, inoportuna o abundante hasta el punto de rebozar el líquido vital en las macetas.

Artefactos que utilizan en el huerto urbano: Alicates de podar para eliminar exceso de follaje o zonas dañadas de las plantas y un rastrillo pequeño para remover el sustrato de las macetas cuando por la falta de riego frecuente se endurecen hasta el punto de retener en su superficie el agua.

Hechos relevantes: Se mencionan los siguientes episodios recordados como anécdotas:

1. Un día la persona de la tercera edad procedió al riego del huerto urbano, al momento que toda la familia estaba de salida, con el apuro se olvida de cerrar la canilla de suministro de agua, al retornar la familia a la casa en un tiempo aproximado de 10 horas, el huerto urbano y gran parte del patio lateral estaban inundados, provocando pérdida del sustrato en las macetas, daños irreparables al cultivo y un impacto económico en la cuenta del servicio de agua potable a fin de mes, además de las quejas de los vecinos por el agua servida en la calle.
2. Por motivo de vacaciones no fue posible realizar el riego o dejar encargado a alguien por 4 días, las plantas se marchitaron con ellas el esfuerzo, tiempo y dinero invertido en el huerto urbano.
3. A veces se realizaba el riego de las plantas por más que las macetas contaban todavía con suficiente humedad en el sustrato, ocasionando la pérdida del sustrato y enfermedades a los cultivos por exceso de humedad (hongos), así como reservorio de agua para la proliferación de insectos.

Técnica de la entrevista estructurada. A partir de la técnica de observación practicada se elaboraron preguntas específicas para la entrevista estructurada pues es un instrumento muy apreciado para obtener respuestas expeditivas de acuerdo al ámbito físico investigado, la entrevista se define como «el arte de escuchar y captar información», (Munch y Ángeles, 1988, p. 61).

Para el efecto fue entrevistado a un especialista en el área, el **Ingeniero Agrónomo Claudio Rufino Cabañas Gálvez**, con cédula de identidad paraguaya número 1.711.025 (36 años de edad), además se suministró al entrevistado un documento impreso (ver Anexo A) con los detalles técnicos del ambiente físico investigado y las preguntas elaboradas teniendo en cuenta a partir de un huerto urbano en macetas con proceso manual de riego por goteo, a continuación se presentan las preguntas y respuestas de la entrevista realizada en fecha 18 de mayo de 2019.

Preguntas teniendo en cuenta las macetas. **1. ¿Qué tipos de cultivos puede albergar, a parte del actual, con esas dimensiones?** Respuesta: En ese tipo de macetas se pueden albergar todos los cultivos hortícolas de porte pequeño a mediano, siendo los tomates (que seguro van a crecer más de 38 cm) y locotes, los más grandes que este sistema puede aguantar, debido al volumen de tierra que puede contener la maceta, siempre pensando en el sistema radicular de la planta tenga un buen desarrollo.

2. ¿Se puede cultivar más de un cultivo, en caso de afirmación cuántos y qué tipos de cultivos son combinables? Respuesta: Si se pueden combinar, depende de que se quiera producir, siempre respetando los espaciamientos recomendados por cada especie y es altamente recomendable la combinación de cultivos en el huerto ya que funciona como control biológico de plagas y enfermedades. Es recomendable, entonces, si se van a combinar cultivos, que no sean de la misma familia botánica. Ej. Tomates más Lechugas más Cebollitas de verdeo, etc.

3. ¿El sustrato de las macetas se deben cambiar por cada nuevo cultivo? Respuesta: No, solo en caso de algún ataque severo de enfermedades o plagas en el huerto. Agregar más humus cada 6 meses, como medida de mantenimiento de los índices de fertilidad del sustrato.

4. ¿Cuál sería el sustrato adecuado para este tipo de macetas en forma general? Respuesta: El sustrato descrito en el trabajo esta correcto y me parece el ideal.

5. ¿Ha realizado alguna vez este tipo de huerto urbano en macetas? Respuesta: Sí, tengo un huerto urbano en mi casa.

6. ¿Tiene algún aporte adicional o recomendaciones sobre este apartado? Respuesta: Creo que la planta del tomate es más grande de lo que el trabajo contempla (las raíces absorben mucha agua del suelo, y más cuando están con frutas), por ende temo que las macetas podrían ser un contenedor de humedad muy variable, hay que calcular que la evapotranspiración del suelo no sea un problema, lo que deriva en dos opciones; macetas más grandes, o, el riego continuo.

Preguntas teniendo en cuenta el riego por goteo **1. ¿Qué cantidad de presión de agua (bares) necesita este tipo de riego por goteo?** Respuesta: La presión es variable a las

condiciones del huerto. Se tienen en cuenta, el caudal de agua que se quiere bombear, la altura del reservorio en relación al terreno, la distancia que se desea trasportar el agua, el material y los accesorios que constituyen la línea de riego y el tipo de picos que se cuenten, en este caso picos por goteo.

2. ¿Qué frecuencia de riego sería el adecuado en la semana y qué duración deberían tener por cada cultivo? Respuesta: Cada cultivo cuenta con sus requerimientos mínimos de humedad en el suelo, el tomate es un cultivo que necesita de mucha agua, especialmente en macetas, cuando está en floración y produciendo frutas.

3. ¿En qué rango de intensidad porcentual de luz solar es recomendable realizar el riego (0 a 100 %)? Respuesta: En mi opinión el riego es más eficiente en horas de menor incidencia solar, pero yo puedo regar de noche y al medio día siguiente, al ver que las macetas están todas secas y las plantas se recienten, entonces se vuelve a regar. El riego es el proceso por el cual aportamos humedad al suelo, de modo a que sea aprovechable por las plantas y se realiza cada vez que sea necesario.

4. ¿En qué rango de temperatura de ambiente en grados Celsius aproximadamente se puede realizar el riego (0 a 50 grados celsius)? Respuesta: Cada cultivo tiene un potencial distinto, algunos son cultivos de invierno, otros de verano y otros se pueden producir todo el año, si se lleva en cuenta el tipo de cultivo con la época del año correspondiente al cultivo, entonces la temperatura del agua se vuelve un dato poco incidente.

5. ¿En qué porcentaje de humedad relativa del suelo es recomendable el riego (0 a 100 %)? Respuesta: Existe un concepto en este sentido que se llama Capacidad de Campo, el mismo explica las variaciones de humedad en el suelo y su relación con las plantas. Pero siempre es necesario aportar humedad al suelo antes que el cultivo llegue al punto de marchitez permanente.

6. ¿Qué cantidad de agua necesita aproximadamente cada maceta (cm³ o litros)? Respuesta: Está directamente relacionado con el tipo de cultivo que alberga la maceta y la época del año. En estas condiciones de macetas y con el Tomate aproximadamente 1500 cm³/día.

7. ¿Se deben tener en cuenta la evapotranspiración en estas macetas para el riego? Respuesta: Sí, es muy importante.

8. ¿Qué cuidados de mantenimiento se deben tener por este tipo de riego por goteo? Respuesta: El riego por goteo es delicado a nivel de los picos de goteo, se quieren trancar por dentro y ya no gotean.

9. ¿Tiene algún aporte adicional o recomendaciones sobre este apartado? Respuesta: Si el agua es de la ESSAP, tener en cuenta el nivel de cloro en el agua, ya que también puede ir trancando los picos de goteo. En este tipo de sistemas de riego es altamente recomendable que la linea de riego cuente con un pre-filtro.

Planteamiento del Problema

En términos ambientales de las zonas verdes urbanas, el cuidado y la preservación del agua es un factor crítico y por tanto para un huerto urbano en macetas sin un dispositivo que combina indicadores ambientales como; lluvia, temperatura ambiente, humedad del suelo y luz solar, además sin la posibilidad de configurar según la especie de cultivo como también la de supervisar en línea tanto visualmente el ambiente físico del huerto y los valores de los indicadores ambientales citados (telemetría) mediante una interfaz de acceso remoto en la automatización de su proceso de riego por goteo, supone un uso ineficiente e irracional del vital líquido en su propósito de mantener el nivel de humedad constante y uniforme en el sustrato de la maceta según necesidad de la hortaliza, por ende deriva en la deficiencia en su desarrollo (crecimiento y nutrición) o pérdida de la misma.

Por lo expuesto se presume que este dispositivo conllevará al ahorro de dinero y tiempo a largo plazo, así como de mejorar el uso adecuado del precioso recurso natural cada vez más escaso: el agua, además de propiciar la motivación para mantener un espacio verde urbano que hoy día adquiere una gran importancia como medio de mejoramiento de la calidad de vida de las personas y de la comunidad misma.

A continuación se presentan las siguientes definiciones en referencia al contexto del tema presentado, Sistema de Control Automático Programable para el Riego por Goteo de un Huerto Urbano en Macetas:

De acuerdo a Mendiburu (2003) se tienen las siguientes definiciones para comprender un **sistema de control automático**:

- *Control*: Acción ejercida con el fin de poder mantener una variable dentro de un rango de valores predeterminados.
- *Sistema de control*: Conjunto de equipos y componentes, que van a permitir llevar a cabo las operaciones de control.
- *Operaciones de control*: Conjunto de acciones que buscan mantener una variable dentro de patrones de funcionamiento deseados.
- *Control automático*: Es el desarrollo de la acción de control, sin la participación directa de un ser humano (operario).
- *Automático*: Es todo aquello que se mueve, regula, y opera, por sí solo, independiente del medio que lo rodea.
- *Automatización*: Consiste de un sistema de control automático, por el cual el sistema verifica su propio funcionamiento, efectuando mediciones y correcciones sin la interferencia del ser humano.
- *Sistemas de automatización*: Conjunto de equipos, sistemas de información, y procedimientos que van a permitir a asegurar un desempeño independiente del proceso, a través de operaciones de control y supervisión.
- *Supervisión*: Es el proceso de lectura de valores de las diversas variables del proceso, con el

objetivo de identificar el estado en el que se viene desarrollando el proceso en un tiempo actual.

En tanto La Real Academia Española (Española, 2003) define las siguientes palabras:

- *Sistema*: Conjunto de reglas o principios sobre una materia racionalmente enlazados entre sí. Conjunto de cosas que relacionadas entre sí ordenadamente contribuyen a determinado objeto.

- *Programable*: Que se puede programar.

- *Programar*: Preparar ciertas máquinas por anticipado para que empiecen a funcionar en el momento previsto.

- *Riego*: Acción y efecto de regar. Agua disponible para regar.

- *Regar*: Esparrcir agua sobre una superficie, como la de la tierra, para beneficiarla, o la de una calle, una sala, etc., para limpiarla o refrescarla.

- *Goteo*: Acción y efecto de gotejar.

- *Gotear*: Dicho de un líquido: Caer gota a gota.

- *Gota*: Partícula de cualquier líquido de forma esferoidal.

- *Huerto*: Terreno de corta extensión, generalmente cercado de pared, en que se plantan verduras, legumbres y a veces árboles frutales.

- *Urbano*: Perteneciente o relativo a la ciudad.

- *Maceta*: Recipiente de barro cocido, que suele tener un agujero en la parte inferior, y que, lleno de tierra, sirve para criar plantas.

Pregunta General

¿Cómo construir un dispositivo con indicadores de lluvia, temperatura ambiente, humedad del suelo y luz solar que permita acceder remotamente al mismo para configurar y supervisar la automatización del proceso de riego por goteo de un huerto urbano en macetas?

Preguntas Específicas

¿Cómo se podrá ensamblar las partes necesarias del hardware del dispositivo?

¿Cómo se desarrollará, instalará y configurará el software para el dispositivo?

¿De qué manera se accederá para configurar y supervisar al dispositivo?

Sustento Teórico

Antecedentes

En la actualidad existe una gran variedad de dispositivos que automatizan el riego de jardines y huertos urbanos, se presentan como referencia tres modelos comerciales:

Nelson modelo 56607. Este modelo básico fue hecho en el año 2011 en los Estados Unidos de América, fabricada por la empresa Fiskars Brands Inc en el estado de Winconsin, es un temporizador electrónico de salida simple externo que utiliza una pantalla de cristal líquido (LCD) para configurar mediante un selector rotativo las funciones de programaciones de riego en cuanto al comienzo, duración y frecuencia mediante botones que van de 1 a 360 minutos de tiempo de ejecución para el riego, además mediante un botón procede a pausar el riego en caso de lluvia posponiéndolo a 24, 48 o 72 horas, también tiene un botón de riego manual con posibilidad de ajustar la duración del riego hasta 240 minutos, su alimentación es a base de 2 pilas alcalinas del tipo AA, no posee conexión a sensores ambientales y su programación se limita a repeticiones en horas y tampoco registra eventos de riegos hechos.

Figura 1: Programador de riego 56607 Nelson. Fuente: Nelson (2011)



Gardena modelo MasterControl. Es modelo intermedio fue fabricado en Alemania por la empresa Gardena miembro del Grupo Husqvarna en el año 2013 para el riego de jardines, se puede escoger hasta seis programas de riego diariamente, la duración del riego puede ajustarse de un minuto a 9 horas y 59 minutos, para mayor flexibilidad, puede escoger libremente los días de riego deseados, si se desea iniciar el riego de forma inmediata, también puede activarse manualmente, la unidad operativa cuenta con una pantalla de grandes dimensiones y de fácil visualización donde puede extraerse para una programación más sencilla. La programación es intuitiva y se explica paso a paso en la pantalla. Puede escogerse entre diez idiomas, además con la conexión de un sensor de lluvia o sensor de humedad Gardena el riego automático se interrumpe en función de la humedad del suelo y la lluvia natural, gracias al sensor de humedad Gardena también se puede controlar el riego basándose únicamente en el sensor, la fuente alimentación es una pila alcalina 9V, este modelo solo admite los tipos de sensores mencionados.

Figura 2: Programador de riego MasterControl Gardena. Fuente: Gardena (2013)



K-rain modelo ProEX 2.0 Wi-Fi Enabled Controller. El modelo de gama alta en la actualidad de fabricación Americana con sede en Florida en el año 2015, es el programador de riego modular que permite expandir fácilmente zonas de riego, su pantalla retroiluminada (la más grande en el mercado), su programación es fácil con sus botones táctiles, elevan al Pro EX 2.0 a un nivel completamente nuevo en el área de los programadores de riego, su control remoto le permite operar al programador justo al alcance de sus manos, los tiempos de riego van de 1 segundo hasta 6 horas para todas las zonas, tiene 3 números de programas, programación de riego en forma personalizado (día de semana), intervalos (1 a 31 días), soporta control remoto de múltiples cuentas desde cualquier lugar y en cualquier momento, no se requieren múltiples controles remotos, puede administrar cuentas a través de celulares inteligentes, tabletas o navegador web, control flexible desde cualquier lugar del mundo, los módulos adicionales son opcionales como control remoto, sensor de lluvia y punto de acceso inalámbrico Wi-Fi, este modelo utiliza alimentación alterna de 115V a 60 Hz.

Figura 3: Programador de riego ProEX 2.0 Wi-Fi Enabled Controller K-Rain. Fuente: K-Rain (2015)



Todos estos sistemas presentan las siguientes limitaciones:

- No poseen franjas de horario de riego más personalizable.
- Si bien algunos presentan algunos indicadores ambientales para regular sus riegos los costos de inversión se maximizan al ir agregando más funcionalidades.
- No cuentan con una unidad de control automatizada programable mediante datos de referencia de indicadores ambientales: lluvia, humedad, temperatura y luz diurna.

- No poseen la funcionalidad de visualización del huerto o jardín en forma remota (foto o video en tiempo real).
- No emiten notificación de alertas.
- No cuentan con datos sobre registros de eventos de riegos realizados o un estimado de cantidad de agua utilizada.
- No poseen una interfaz de usuario remota para la supervisión, programación de riego y personalización de datos de referencia (domotización).

Bases teóricas

Para el presente proyecto he recabado y analizado informaciones técnicas sobre el ambiente físico investigado con respecto al huerto urbano, el cultivo en macetas y el riego por goteo, posteriormente he procedido a investigar las ciencias de ingeniería aplicada sobre sistemas de control y de automatización, teniendo en cuenta éstos principios he indagado sobre los fundamentos de la electricidad, la electrónica y la informática para comprender su aplicación combinada en un sistema embebido o empotrado en tiempo real, y con ésta base de conocimientos, he escudriñado en páginas web de distribuidores de productos de electrónica de acuerdo a las partes requeridas para el dispositivo propuesto: placas bases de hardware abierto, módulos electrónicos estandarizados, periféricos de entrada/salida y fuentes de alimentación de energía. Por último mediante las referencias técnicas proporcionadas por los fabricantes de cada parte del dispositivo se obtuvo un cuadro comparativo y la manera de configurar, adaptar e interactuar con software libres compatibles, con el objetivo de desarrollar el firmware de control/automatización como también el software de interfaz gráfica del usuario final para la configuración y supervisión del dispositivo en forma remota en el proceso de riego por goteo de un huerto urbano en macetas.

Huerto urbano. Teniendo en cuenta a de la Vega (2011) hoy en día la mayor parte de la población vive en las ciudades, en un ambiente urbano, cargado de asfalto, ruido y estrés. Los paisajes y la actividad agrícola urbana y peri-urbana se ha reducido ostensiblemente, aumentando los espacios dedicados a la industria y a los servicios. Nuestro contacto con la Naturaleza se ha reducido en muchas ocasiones a los parques y jardines urbanos o a salidas esporádicas al campo los fines de semana. Este alejamiento del mundo natural y rural hace que al urbanista le cueste mucho entender los ciclos de la Naturaleza y los cambios estacionales. Además está acostumbrado a acceder a todo tipo de alimentos de una forma inmediata y en cualquier época del año, sin pararse a analizar de donde vienen estos alimentos ó en qué condiciones se han producido. Sin embargo, cada vez son más las personas que demandan el reverdecimiento de las ciudades, la recuperación de la agricultura urbana y peri-urbana y la preocupación por acceder a alimentos frescos y ecológicos. En este contexto, crear un pequeño huerto en casa, en un balcón, terraza o patio, se convierte en una actividad que puede ser muy

satisfactoria y enriquecedora. De hecho cuanto más urbano es el paisaje que nos rodea más éxito tendrá esta experiencia que nos permitirá entender mejor la Naturaleza y valorar la contribución de la agricultura tradicional y ecológica al desarrollo sostenible.

También Schonwald y Pescio (2015) indica que la producción propia de alimentos ha cobrado relevancia pública en el mundo y constituye un compromiso con nuestro bienestar y la apuesta por un futuro más equitativo. Asimismo, la producción de alimentos en la ciudad favorece el uso eficiente del agua, la energía y el suelo y ayuda a ahorrar energía, ya que productores y consumidores están más cerca entre sí.

En el caso de la ciudad, tanto el deterioro de los suelos como la falta de espacio son factores que alentaron la búsqueda de alternativas de producción diferentes del cultivo tradicional sobre suelo. En esa línea, las huertas en macetas también permiten reutilizar muchos materiales inorgánicos y orgánicos que suelen desecharse en los hogares. Y hasta, incluso, se presentan como un medio para recuperar los saberes de nuestros antecesores y compartirlos con las nuevas generaciones. De este modo, la producción de alimentos agroecológicos en nuestra casa significa que podamos ser responsables del sustento propio.

Beneficios. Los beneficios de acuerdo a de la Vega (2011):

- Potencia la capacidad de observación y de entendimiento del medio natural.
- Aumenta la sensibilidad hacia la sostenibilidad.
- El huerto resulta una actividad muy divertida, relajante y que disminuye el estrés.
- El huerto es una herramienta extraordinaria para la educación ambiental de nuestros hijos.
- Permite redescubrir la calidad organoléptica de los alimentos.
- Seguridad de que lo se come no lleva ningún plaguicida ni herbicida de síntesis química, ya se ha cultivado siguiendo prácticas ecológicas.
- Motiva a preocupar sobre la calidad de los alimentos que se compra.
- Ayuda a valorizar la figura del agricultor y la agricultura tradicional.
- Además en el caso de huertos realizados en terrazas o azoteas puede llegar a tener importancia en el ahorro energético de la vivienda.

Todos estos y otros muchos son beneficios que puede aportar tener un pequeño huerto en casa, siempre que se tenga claros los objetivos del mismo y se supere las dificultades iniciales que, como personas de ciudad, se puede tener: impaciencia, querer obtener resultados rápidamente y sin complicaciones, poca capacidad de observación, querer reproducir la agricultura convencional en nuestro balcón, etc.

Una vez superad estos inconvenientes y adoptando una actitud de curiosidad y de predisposición al aprendizaje continuo, se consigue que el huerto sea una experiencia fascinante.

Cultivo en macetas. Según Luaces (2018) cualquier contenedor puede ser utilizado para producir alimentos, siempre y cuando no haya almacenado sustancias tóxicas o nocivas. Es

fundamental realizar perforaciones en la base del envase para favorecer el drenaje del agua de riego. La acumulación de agua produce la asfixia de las raíces de las plantas.

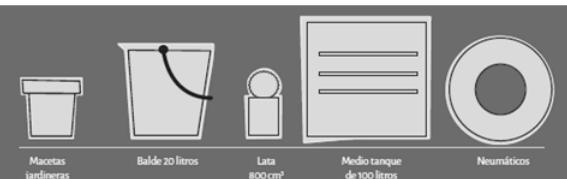
El sustrato. Teniendo en cuenta a Luaces (2018) el cultivo en envases ofrece la ventaja de combinar, en las proporciones adecuadas, los materiales que utilizaremos para el sustrato. Un buen sustrato aporta nutrientes a las plantas, retiene las cantidades necesarias de agua y drena el exceso de humedad. Para lograrlo, se mezcla: 1 parte de tierra negra, 3 partes de abono orgánico maduro, 1 parte de arena gruesa, viruta o cascarilla de arroz.

El contenedor adecuado. De acuerdo a Oncina y Castelló (2013) cuando una planta se cultiva en maceta se debe tener en cuenta que se enfrenta a condiciones desfavorables como:

- Un volumen reducido de tierra donde hacer crecer sus raíces, pudiendo existir descompensaciones entre la parte aérea y subterránea de la planta.
- Los cambios de temperatura (frío/calor) se intensifican en la maceta.
- El agua que puede almacenar la tierra del macetero es muy limitada y se evapora de una forma mas o menos rápida según la época del año, tamaño de la planta, mezcla del sustrato, tamaño y material del macetero

Por lo tanto se debe tener en cuenta que con el cultivo en macetas se está sometiendo a las plantas a condiciones en un medio desfavorable para su desarrollo.

Tabla 1: Recipientes adecuados según cultivo. Fuente: Schonwald y Pescio (2015)




Especies	Envase sugerido	Observaciones
Radicheta, rúcula, cilantro, espinaca, ciboulette, perejil	Macetas jardineras Latas de 800 cm³	Se siembran a poca profundidad.
Lechuga, acelga, apio, copete, frutilla	Macetas jardineras Latas de 800 cm³	Coloque una planta por envase.
Zanahoria, rabanito, remolacha	Macetas jardineras	Al igual que para el perejil y la espinaca, respese la densidad de siembra.
Zapallo, melón, sandía	Balde de 20 l	Requieren bastante volumen de tierra. Se guian por paredes y techos.
Tomate	Común: Balde de 20 l Cherry: Latas de 800 cm³ y macetas jardineras	Necesita un mínimo de cinco horas de sol por día.
Arvejas, chuchas	Macetas jardineras	Al ser trepadoras, utilice tutores o una malla plástica como espaldadera.
Cebolla, ajo, puerro	Macetas jardineras	Su cosecha puede demorar hasta ocho meses.
Berenjena, pimiento	Macetas jardineras	Deben recibir buena luz diaria y colocarse una planta por maceta.
Papa	Neumáticos	Junto con la zanahoria, la papa se adapta mejor a los suelos arenosos.
Limón, quinoto, higo, ciruela	Medio tanque (100 l)	Para florecer y fructificar, requieren abundante sol.

Cultivo	Diámetro maceta (en cm)	Volumen (litros)
Alcachofa	50	45-50
Berenjena	Melón	
Calabacín		
Calabaza	Sandía	40
Acelga	Habas	30-45
Brócoli	Nabo	
Col	Pimienta	
Coliflor	Tomate	30
Apio	Pepino	20-30
Boniatto	Patata	
Judía	Maíz	10-20
Espinaca	Perejil	
Guisante		
Lechugas	Remolacha	20
	Cebolla	5
Ajo	Zanahoria	3
Puerro	Rabanitos	10
		2

Riego por goteo o riegos localizados a baja presión. De acuerdo a Alba y Labajos (2017) si se diseñan de forma conveniente y se utiliza la experiencia para saber con qué frecuencia y abundancias se tiene que regar, el ahorro de agua puede ser muy significativo y puede llegar a superar un 30%, pero hay revisar a menudo, observar la humedad del suelo del huerto con frecuencia. Hay muchos tipos diferentes de emisores (goteros pinchados, insertados o integrados, microdifusión, microaspersión, mangueras de exudación), pero siempre que sea posible se debe instalar riegos localizados por goteo ya que permite trabajar a presiones bajas, incluso casi a presión 0, siendo muy apto para instalaciones pequeñas que precisan solo 1 a 2 Kg./cm² de presión, el poder trabajar a muy bajas presiones permite instalar riego por goteo desde bidones o pequeños depósitos situados en las zonas altas del microhuerto urbano ó elevados artificialmente. Para huertos mas grandes si no tenemos la mínima presión requerida, se debe tener grupos de bombeo que nos den presión o bien tomas de agua conectadas a redes de agua a presión, se debe asegurar midiendo con manómetro en las tomas. El riego localizado por goteo, tiene además una alta uniformidad del riego (90 a 95 %) y además, al no mojar todo el suelo, evita la proliferación de hierbas no deseadas y de problemas fitosanitarios y se puede automatizar y controlar fácilmente.

Componentes del riego localizado por goteo. De acuerdo a Alba y Labajos (2017) se van a distinguir distintas partes en una instalación de riego localizado por goteo:

El cabezal de riego, compuesto por los mecanismos que se colocan en el punto de agua al comienzo de la instalación.

Suelen incluir para cada sector de riego:

- Un programador en el que señalamos a qué hora queremos que se abran los riegos y a qué hora deben cerrarse.
- Una electroválvula: que se abre para dejar pasar el agua o se cierra, a las horas que le dice el programador.
- Un filtro que evita que lleguen impurezas en el agua que puedan obstruir los goteros o dejar abierta la electroválvula.
- Un reductor de presión que reduce la presión de la toma de agua en el caso de que sea muy elevada, para que los goteros puedan funcionar bien o no salten las uniones.

En huertos, que suelen ser pequeños y abastecidos con agua potable de la red local de servicio de agua potable, estos mecanismos se suelen colocar dentro de una caja en el suelo denominada “arqueta”, de la arqueta salen las tuberías generalmente enterradas que llevan el agua hasta los puntos donde están los cultivos (los bancales, eras de cultivo, frutales, zonas de aromáticas, plantas de jardinería). Estas tuberías suelen ser de polietileno.

Una vez que llegan estas tuberías a las zonas de cultivo, abastecerán a los “ramales” que son las tuberías portagoteros. Los goteros son los emisores por donde sale el agua que va a regar a los cultivos, estas tuberías son mucho más finas (6 mm de diámetro habitualmente) y van en la superficie del suelo, no enterradas. La elección que se haga de goteros y la distribución de los

ramales va a determinar todo el diseño hidráulico de la instalación de riego.

Algunos conceptos básicos de hidráulica. Teniendo en cuenta a Alba y Labajos (2017) a la hora de diseñar la instalación se tiene que tener en cuenta distintos conceptos hidráulicos:

- *Caudal*: es la cantidad de agua por unidad de tiempo. Se necesita saber el caudal que hay en el punto de agua, si se puede medir directamente usando un cubo de un volumen conocido y midiendo con un cronómetro cuanto tarda en llenarse. O también se puede estimar con tablas en función del diámetro de la toma.

- *Presión*: es la fuerza ejercida sobre una superficie determinada. Se tiene que dimensionar la instalación de riego de forma que, la presión en el punto de agua sea suficiente para que los goteros rieguen adecuadamente (les llegue como mínimo el agua a la presión nominal de funcionamiento), que el agua supere los desniveles del terreno que haya en la parcela; y que el agua supere la perdida de carga, que es la perdida de presión debido al rozamiento del agua con las tuberías y piezas especiales.

En la toma de agua, si es posible, se medirán dos tipos de presión:

- *Presión estática*: es la que tiene la toma de agua cuando esta cerrada (cuando no funciona el riego). Nos valdrá para determinar el timbre de las tuberías.

- *Presión dinámica*: es la que tiene la toma de agua cuando esta funcionando el riego, cuando esta abierta la llave de paso.

Diseño de una instalación de riego. Según Alba y Labajos (2017) a la hora de diseñar una instalación de riego localizado, es interesante recopilar una serie de información previamente y tomar una serie de decisiones. Estas se podrían concretar de la siguiente manera:

- *Suelo*: tenemos que saber, aproximadamente, si nuestro suelo es mas bien arenoso, mas bien arcilloso o una mezcla de ambos, también si queremos calcular nosotros las necesidades brutas de agua de los cultivos tenemos que recopilar algunos datos climáticos (precipitación efectiva y evapotranspiración potencial) para la localidad donde este el huerto o en sus proximidades, muy importante es tener un plano o croquis del huerto, donde se indiquen los tipos de cultivos y las superficies que ocupan aproximadamente (bancales o eras de hortícolas, zonas de aromática o de plantas de jardinería, frutales, setos, etc.), en el caso de macetas se debe conocer el volumen del contenedor para determinar la capacidad del campo, en forma experimental se riega la maceta hasta que el agua se filtre por el agujero de la misma para determinar la cantidad de agua retenida por el sustrato.

- *Agua*: donde está ubicada, que caudal y que presión estática y dinámica tiene.

- *Gotero*: el caudal de los goteros que se va ha instalar, la separación de goteros en el mismo ramal que se ponga y la separación entre ramales que va ha tener.

Con toda esta información recopilada, ya podremos calcular varias cosas:

- Cálculo del intervalo máximo entre riegos.

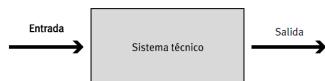
- Tiempo de riego, la misma se calcula de la siguiente forma:

$$\text{Tiempo de Riego (minutos)} = \frac{\text{Necesidades Brutas (litros/m}^2\text{)}}{\text{Caudal del emisor (litros/hora)}} \times \frac{1}{\text{Nº de emisores por m}^2} \times 60$$

$$\text{Nº de emisores por m}^2 = \frac{1}{\text{Distancia entre emisores (metros)} \times \text{distancia entre ramales (metros)}}$$

Fundamentos de las ciencias de ingeniería. Las ciencias de ingeniería según (Ebel *et al.*, 2007) al igual que las ciencias humanísticas, sociales y naturales, son una disciplina académica propia y, por lo tanto, tienen sus propios conceptos técnicos, procedimientos y medios. Se sobreentiende que se basan en otras ciencias y las utilizan, especialmente las matemáticas y la física, aunque también las ciencias sociales. Sin embargo, estas otras ciencias también recurren en buena medida a los resultados de la investigación realizada en el ámbito de las ciencias de ingeniería. A diferencia de las ciencias naturales, las ciencias de ingeniería no se dedican únicamente al descubrimiento de leyes naturales; su objetivo más bien consiste en encontrar soluciones técnicas para satisfacer necesidades del ser humano. Por lo tanto, el ingeniero siempre se pregunta primero cómo solucionar un problema determinado. Esta actitud redundante en formas de trabajo que son típicas en las ciencias de ingeniería, como la forma de pensar en cajas negras. Ello significa que utilizan sistemas técnicos para sus fines, sin que tengan que saber exactamente cómo funcionan sus partes. Al ingeniero le basta saber que un aparato que recibe un “input” (entrada) determinado ofrece un “output” (salida) determinado.

Figura 4: La caja negra de un sistema técnico. Fuente: Ebel *et al.* (2007)



Teniendo en cuenta a (Ebel *et al.*, 2007) un ejemplo práctico sería en máquinas automáticas se utilizan motores eléctricos de diversos tamaños y de potencias diferentes. El ingeniero encargado del diseño de una máquina no tiene que saber cómo funciona exactamente el motor eléctrico. Es suficiente que sepa seleccionar el motor apropiado en función de diversos parámetros como dimensiones, momento de giro, revoluciones, consumo de corriente, potencia, etc. La situación es diferente, claro está, si el ingeniero está a cargo de la construcción de motores eléctricos. En ese caso, debe disponer de conocimientos detallados sobre el funcionamiento y las características físicas del motor eléctrico y de todas sus partes. Otra característica de las ciencias de ingeniería consiste en la forma de representar las soluciones técnicas. Los ingenieros y técnicos recurren a medios descriptivos estandarizados, que se entienden en todo el mundo y que, por lo general, son medios gráficos. Los más importantes son los siguientes: dibujos técnicos y listas de piezas, esquemas de distribución, diagramas de flujo y programas, esquemas técnicos y diagramas esquemáticos.

Ejemplos de especialidades de la ciencia de ingeniería: ingeniería mecánica, ingeniería eléctrica, ingeniería de las técnicas de fabricación, ingeniería de la construcción, ingeniería electrónica, ingeniería informática, entre otras, todas tienen en común la investigación, la obtención de soluciones técnicas y su aplicación y se distinguen por la materia que tratan y por la orientación de cada especialidad.

Según (Ebel *et al.*, 2007) la técnica de la automatización es una disciplina que abarca varias especialidades y que por lo tanto, recurren a conocimientos y métodos de diversas ciencias de ingeniería.

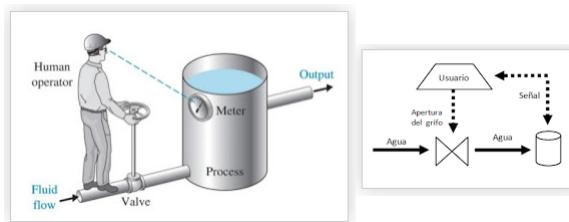
Sistemas de control.

Manual. De acuerdo a Estefania (2017) en nuestra vida diaria manejamos diversas formas de energía a nuestra voluntad, encendemos luces, regulamos la llama de una hornalla, abrimos y cerramos canillas, disminuimos el volumen del televisor, giramos el volante de un vehículo, lo aceleramos, lo frenamos. En todos estos casos nosotros formamos parte de un sistema de control. Nuestro rol es darle información a estos sistemas.

Hablamos de control manual toda vez que existe la presencia y la intervención de una persona en la acción de controlar y regular el comportamiento del sistema. Esta persona participa en forma activa, registrando la inspección a través de sus sentidos (vista, olfato, etc.) y actuando con sus manos u otra parte del cuerpo, para llevar al sistema hacia los valores normales.

El objetivo de los sistemas de control es mantener en un determinado valor (o margen de valores) las variables de un proceso. Por ejemplo, temperatura, presión, caudal, nivel.

Figura 5: Ejemplo control manual. Fuente: (Estefania, 2017)

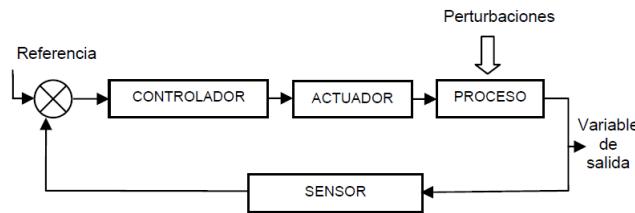


Automático. Según Estefania (2017), un sistema de control se vuelve automático si se usa un elemento llamado controlador que opere en reemplazo del operario humano. La función del controlador será, a partir de ciertos datos, decidir qué acción de control será necesaria y realizar, en consecuencia, la regulación para mantener el sistema en determinados valores o estados.

Elementos de un sistema de control automático. Mendiburu (2003) afirma que el sistema de control va a actuar independiente del operario y va a determinar por sí mismo los mejores valores para las señales de control. Para ello se contará con una referencia, que es un valor dado

por el operario, este valor es fijo y depende del tipo de proceso y de las exigencias que este amerite; es conocido como *set-point*, este valor es el que se desea alcanzar y mantener.

Figura 6: Diagrama en bloques de los elementos del sistema de control automático. Fuente: Mendiburu (2003)



Así tenemos los cuatro elementos que conforman el sistema de control según Mendiburu (2003):

- *Controlador (unidad de control)*: es aquel instrumento que compara el valor medido con el valor deseado, en base a esta comparación calcula un error (diferencia entre valor medido y deseado), para luego actuar a fin de corregir este error . Tiene por objetivo elaborar la señal de control que permita que la variable controlada corresponda a la señal de referencia. Los controladores pueden ser de tipo manual, neumático, electrónico; los controladores electrónicos más usados son: computadoras con tarjetas de adquisición de datos, PLC (controladores lógicos programables), microcontroladores (PIC) y placas Arduino. En resumen el controlador es el dispositivo que se utiliza para almacenar y ejecutar una serie de instrucciones (el programa de control) activando o desactivando actuadores de acuerdo con los datos que brindan los sensores.

- *Actuador (salidas)*: es aquel equipo que sirve para regular la variable de control y ejecutar la acción de control, es conocido como elemento final de control, estos pueden ser de 3 tipos:

a. Actuadores eléctricos: son usados para posicionar dispositivos de movimientos lineales o rotacionales. Ejemplo: motor, relé, switch, electroválvulas.

b. Actuadores neumáticos: trabajan con señales de presión, estas señales son convertidas a movimientos mecánicos. Ejemplo: pistones neumáticos, válvulas.

c. Actuadores hidráulicos: operan igual a los neumáticos, son usados en tareas que requieren mayor fuerza por ejemplo levantar compuertas, mover grúas, elevadores, etc. Ejemplo: pistones hidráulicos.

Por lo tanto los actuadores son dispositivos que se utilizan para generar los efectos buscados a través de un sistema de control, convierten en acciones las órdenes que reciben de los controladores.

- *Proceso*: está referido al equipo que va a ser automatizado, por ejemplo puede ser una bomba, tolva, tanque, compresor, molino, intercambiador de calor, horno, secador, electroválvula, caldera, etc.

- *Sensor (entradas)*: es un elemento de medición de parámetros o variables del proceso. Los sensores pueden ser usados también como indicadores, para transformar la señal medida en

señal eléctrica. Los sensores más comunes son los de nivel, temperatura, presencia, proximidad, flujo, presión, entre otros. Pueden ser de varios tipos:

Sensores de contacto: son aquellos que realizan la medida en contacto directo, real y físico con el producto o materia. Ejemplo: sensores de boya para medir nivel en un tanque, termocupla para medir temperatura, etc.

Sensores de no contacto: se basan en propiedades físicas de los materiales, son más exactos, pero propensos a interferencias del medio ambiente. Ejemplo: sensores ultrasónicos, sensores ópticos, etc.

Sensores digitales: trabajan con señales digitales, en código binario, pueden representar la codificación de una señal analógica, o también la representación de dos estados apagado/encendido. Ejemplo: sensores tipo switch.

Sensores analógicos: proporcionan medidas continuas, los rangos típicos son de 0 a 20 mA, 4 a 20 mA, 0 a 5 V, 1 a 5 V, entre otros. Ejemplo: sensores capacitivos, sensores piezoresistivos, etc.

Sensores mecánicos: son aquellos que traducen la acción física del elemento medido, en un comportamiento mecánico, típicamente de movimiento y/o calor. Ejemplo: barómetro, termómetro de mercurio, etc.

Sensores electro-mecánicos: este tipo de sensor emplea un elemento mecánico elástico combinado con un transductor eléctrico. Ej.: sensores resistivos, sensores magnéticos, entre otros. En definitiva los sensores son dispositivos que se utilizan para tomar datos de la realidad sobre la que se pretende intervenir, reemplazan los sentidos (tacto, vista, oído, olfato y gusto) del usuario, pero también pueden superar la capacidad de los sentidos, detectando variaciones que las personas no podrían percibir como rayos ultravioletas, ultrasonidos, etc., con lo que se logra mayor precisión en los procesos.

Tipos de sistemas de control.

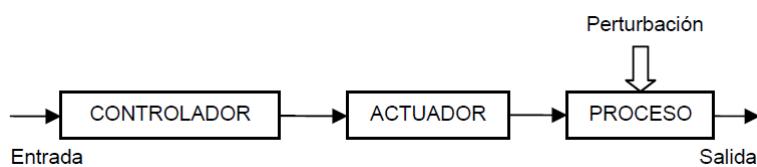
Lazo abierto. Según Ogata (2010) los sistemas en los cuales la salida no tiene efecto sobre la acción de control se denominan sistemas de control en lazo abierto. En otras palabras, en un sistema de control en lazo abierto no se mide la salida ni se realimenta para compararla con la entrada. Un ejemplo práctico es una lavadora. El remojo, el lavado y el centrifugado en la lavadora operan con una base de tiempo. La máquina no mide la señal de salida, que es la limpieza de la ropa.

En cualquier sistema de control en lazo abierto, la salida no se compara con la entrada de referencia. Así, a cada entrada de referencia le corresponde una condición de operación fija; como resultado de ello, la precisión del sistema depende de la calibración. Ante la presencia de perturbaciones, un sistema de control en lazo abierto no realiza la tarea deseada. En la práctica, el control en lazo abierto sólo se usa si se conoce la relación entre la entrada y la salida y si no hay perturbaciones internas ni externas. Es evidente que estos sistemas no son de control

realimentado. Obsérvese que cualquier sistema de control que opere con una base de tiempo está en lazo abierto. Por ejemplo, el control de tráfico mediante señales operadas con una base de tiempo es otro ejemplo de control en lazo abierto. Las ventajas fundamentales de los sistemas de control en lazo abierto son las siguientes: construcción simple y facilidad de mantenimiento, menos costosos que el correspondiente sistema en lazo cerrado, no hay problemas de estabilidad y convenientes cuando la salida es difícil de medir o cuando medir la salida de manera precisa no es económicamente viable. (Por ejemplo, en el caso de la lavadora, sería bastante costoso proporcionar un dispositivo para medir la calidad de la salida de la lavadora, es decir, la limpieza de la ropa lavada.)

Las desventajas fundamentales de los sistemas de control en lazo abierto son las siguientes: las perturbaciones y los cambios en la calibración originan errores, y la salida puede ser diferente de lo que se desea, para mantener la calidad requerida en la salida, es necesaria la recalibración de vez en cuando.

Figura 7: Diagrama en bloques del sistema de control en lazo abierto. Fuente: Mendiburu (2003)

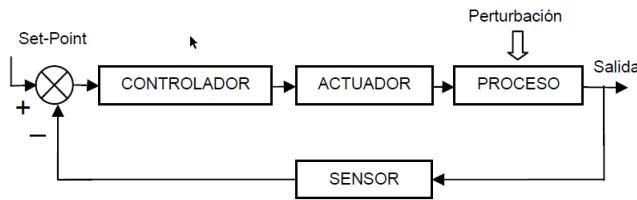


Lazo cerrado. Según Ogata (2010) los sistemas de control realimentados se denominan también sistemas de control en lazo cerrado. En la práctica, los términos control realimentado y control en lazo cerrado se usan indistintamente. En un sistema de control en lazo cerrado, se alimenta al controlador la señal de error de actuación, que es la diferencia entre la señal de entrada y la señal de realimentación (que puede ser la propia señal de salida o una función de la señal de salida y sus derivadas y/o integrales), con el fin de reducir el error y llevar la salida del sistema a un valor deseado. El término control en lazo cerrado siempre implica el uso de una acción de control realimentado para reducir el error del sistema. Una ventaja del sistema de control en lazo cerrado es que el uso de la realimentación vuelve la respuesta del sistema relativamente insensible a las perturbaciones externas y a las variaciones internas en los parámetros del sistema. Es así posible usar componentes relativamente poco precisos y baratos para obtener el control adecuado de una planta determinada. Por otra parte, la estabilidad es un gran problema en el sistema de control en lazo cerrado, que puede conducir a corregir en exceso errores que producen oscilaciones de amplitud constante o cambiante. Debe señalarse que, para los sistemas en los que se conocen con anticipación las entradas y en los cuales no hay perturbaciones, es aconsejable emplear un control en lazo abierto. Los sistemas de control en lazo cerrado sólo tienen ventajas cuando se presentan perturbaciones y/o variaciones impredecibles en los componentes del sistema.

Obsérvese que la potencia nominal de salida determina en forma parcial el coste, peso y

tamaño de un sistema de control. El número de componentes usados en un sistema de control en lazo cerrado es mayor que el que se emplea para un sistema de control equivalente en lazo abierto. Por tanto, el sistema de control en lazo cerrado suele tener costes y potencias más grandes. Por lo general, una combinación adecuada de controles en lazo abierto y en lazo cerrado es menos costosa y ofrecerá un comportamiento satisfactorio del sistema global.

Figura 8: Diagrama en bloques del sistema de control en lazo cerrado. Fuente: Mendiburu (2003)



Clasificación de los sistemas de control. Teniendo en cuenta a Mendiburu (2003) los sistemas de control pueden ser clasificados, basándose en varios criterios, así pues, se puede tener las siguientes clasificaciones según:

Dimensión. - *Sistemas de parámetros concentrados*: son aquellos que pueden ser descritos por ecuaciones diferenciales ordinarias. También son conocidos como sistemas de dimensión finita.

- *Sistemas de parámetros distribuidos*: son aquellos que requieren ecuaciones en diferencia (ecuaciones diferenciales con derivadas parciales). También son conocidos como sistemas de dimensión infinita.

Conocimiento de sus parámetros. - *Sistemas determinísticos*: en estos sistemas se conocen exactamente el valor que corresponde a los parámetros. Por ejemplo un circuito RLC encargado de suministrar tensión a un equipo.

- *Sistemas estocásticos*: en este caso, la forma de conocer algunos o todos los valores de los parámetros, es por medio de métodos probabilísticos. Por ejemplo un horno o caldero que ha acumulado sarro y otras impurezas (las cuales no tienen una función matemática conocida).

Carácter de transmisión en el tiempo. - *Sistemas continuos*: son aquellos descritos mediante ecuaciones diferenciales, donde las variables poseen un valor para todo tiempo posible dentro de un intervalo de tiempo finito. Está referido a las señales analógicas, y su comportamiento matemático es similar a una onda continua. Por ejemplo un proceso de llenado de balones de gas.

Sistemas discretos: son aquellos descritos mediante ecuaciones en diferencia, y solo poseen valores para determinados instantes de tiempo, separados por intervalos dados por un período

constante. Está referido a las señales digitales, y su comportamiento matemático es similar a un tren de pulsos. Por ejemplo el encendido y apagado de un switch que acciona una alarma.

Presencia de linealidad. - *Sistemas lineales:* son aquellos cuyo comportamiento está definido por medio de ecuaciones diferenciales lineales, es decir, los coeficientes son constantes o funciones de la variable independiente. Deben cumplir con el principio de superposición. Por ejemplo un amplificador de señales.

- *Sistemas no lineales:* en caso de que una o más de las ecuaciones diferenciales no se alinean, todo el sistema será no lineal. También se considerará como sistema no lineal a aquellos para los cuales el principio de superposición no sea válido. Por ejemplo el calentamiento de un horno.

Comportamiento en el tiempo. - *Sistemas invariantes en el tiempo:* ocurre cuando todos sus parámetros son constantes, y por tanto se mantiene en un estado estacionario permanentemente. Se define por ecuaciones diferenciales cuyos coeficientes son constantes. Por ejemplo la mezcla de sustancias dentro de un tanque que siempre contiene la misma cantidad y tipo de elementos.

- *Sistemas variantes en el tiempo:* ocurre cuando uno o más de sus parámetros varía en el tiempo, y por lo tanto no se mantiene en estado estacionario. Se define por ecuaciones diferenciales cuyos coeficientes son funciones del tiempo. Por ejemplo para un motor de un vehículo de carrera, la masa del vehículo va a variar por acción del consumo de combustible.

- *Sistemas numéricos:* esta referido a sistemas de control que almacenan información numérica, la cual incluye algunas variables del proceso codificadas por medio de instrucciones. Por ejemplo tornos, taladros, esmeriles, los cuales almacenan información referente a posición, dirección, velocidad, etc.

Características del control. De acuerdo a Mendiburu (2003) existe formas y métodos a través de los cuales los sistemas de control pueden ser representados por medio de funciones matemáticas, esta representación recibe el nombre de Modelamiento Matemático, este modelo describirá las características dinámicas del sistema a través de ecuaciones diferenciales. El modelamiento puede ser:

- *Analítico:* cuando se aplica las leyes físicas correspondientes a cada componente del sistema, que en conjunto forman una estructura o función matemática.

- *Experimental:* consiste en la identificación de los parámetros, mediante el análisis de datos de entrada y salida, estimando valores posibles que se ajusten al sistema a partir del modelamiento matemático, aplicando formulas matemáticas, teoremas, y transformadas, se puede llegar a una función que represente la relación entre la salida y entrada del sistema, esta función se denomina Función de Transferencia. El proceso experimental es denominado Identificación de Sistemas, y corresponde a la planta o proceso que se desea analizar, consiste en recoger datos de la variable de salida con su correspondiente dato de entrada que provocó dicha salida, para luego mediante algoritmos matemáticos aproximar una función de transferencia,

la cual debe general una salida (estimada) similar a la salida censada, y dependiendo de la diferencia entre ambas (error) se dará validez a la función obtenida, o se tendrá que recalcular con nuevos valores en los algoritmos matemáticos de análisis. El análisis de un sistema que se desea controlar, significa analizar su comportamiento dinámico en el tiempo, partiendo de sus características matemáticas se puede llegar a conclusiones respecto al funcionamiento del sistema, tanto aislado como dentro de un lazo cerrado, afectado por ruido y gobernado por un controlador.

Para conocer dicho funcionamiento se debe llegar a conclusiones puntuales respecto a las siguientes características.

- *Estabilidad*: se dice que un sistema es estable cuando después de transcurrido un tiempo t , su valor de respuesta (salida) permanece constante. A este tiempo se le denomina tiempo de establecimiento (time setting), y al valor alcanzado se le denomina valor en estado estable (steady state value), el cual puede ser un valor oscilante dentro de un margen porcentual mínimo, definido a criterio del programador. Un sistema se considera inestable cuando su respuesta luego de transcurrido un tiempo t se mantiene oscilando, variando entre un rango de valores periódicos o simplemente se obtiene cualquier valor aleatorio.

- *Exactitud*: la exactitud del sistema se mide en base a la desviación existente entre el valor deseado (referencia) y el valor real obtenido en la respuesta del sistema (valor en estado estable), a esta diferencia se le denomina error en estado estable.

- *Velocidad de respuesta*: esta característica indica que tan rápidamente es capaz de llegar el sistema, a su valor en estado estable o estacionario.

- *Sensibilidad*: este concepto explica la dependencia de unas variables con respecto a otras, puesto que en un sistema habrá algunas variables manipuladas, otras controladas, y otras perturbadoras, es inevitable que la acción de una repercuta sobre las otras, por ello la necesidad de conocer e identificar cada variable a fin de conocer su naturaleza antes mencionada.

- *Alcanzabilidad*: cuando un sistema cuenta con esta característica, entonces mediante un controlador se puede llevar este sistema desde un estado inicial hasta otro estado cualquiera, en un tiempo finito.

- *Controlabilidad*: un sistema es controlable cuando es posible llevar al sistema a una posición de equilibrio, al aplicarle una entrada y transcurrido un período de tiempo finito.

- *Observabilidad*: se dice que un sistema es de rango completo observable, si mediante la observación de la salida y es posible determinar cualquier estado $x(t)$, en un tiempo finito.

Métodos de control. Mendiburu (2003) explica que existen métodos y estrategias para realizar la acción de control, los métodos de control (clásico y moderno) permiten al controlador reaccionar mandando una señal correctiva del error, mientras que las estrategias de control avanzado hacen más eficiente a la labor de control, ahorrando recursos y tiempo.

Clásico. Los métodos de control clásico son aquellos que esperan a que se produzca un error para luego realizar una acción correctiva. El error se presenta a causa de la diferencia de lectura entre la variable de salida censada y la señal de referencia, este error está presente en todo momento, y la finalidad es minimizarlo. En algunos casos suele generarse un comportamiento oscilatorio alrededor del valor de referencia. Los métodos de control clásico pueden ser:

- *Control on-off*: este método solo acepta dos posiciones para el actuador: encendido (100%) y apagado (0%). La lógica de funcionamiento es tener un punto de referencia, si la variable es mayor el actuador a sume una posición, y si la variable es menor el actuador a sume la otra posición. Por ejemplo tenemos los sistemas de seguridad contra robos, las refrigeradoras domésticas, sistemas de aire acondicionado, etc.

- *Controlador proporcional (P)*: es un control que se basa en la ganancia aplicada al sistema, se basa en el principio de que la respuesta del controlador debe ser proporcional a la magnitud del error. No corrige ni elimina perturbaciones, puede atenuar o aumentar la señal de error. Se representa a través del parámetro K_p y define la fuerza o potencia con que el controlador reacciona frente a un error.

- *Controlador integral (I)*: conocido como RESET. Este tipo de controlador anula errores y corrige perturbaciones, mediante la búsqueda de la señal de referencia, necesita de un tiempo T_i para localizar dicha señal. Se representa mediante el término K_i que es el coeficiente de acción integral y es igual a $1/T_i$

- *Controlador derivativo (D)*: conocido como RATE. Este controlador por sí solo no es utilizado, necesita estar junto al proporcional y al integral. Sirve para dar la rapidez o aceleración a la acción de control. Necesita de una diferencial de tiempo T_d para alcanzar la señal de referencia, se representa mediante el término K_d que es el coeficiente de acción derivativa y es igual a $1/T_d$.

- *Controlador proporcional-integral (PI)*: actúa en forma rápida, tiene una ganancia y corrige el error, no experimenta un offset en estado estacionario. La aplicación típica es en el control de temperatura. Función de Transferencia: $sT_i K_p + 1$

- *Controlador proporcional-derivativo (PD)*: es estable, y reduce los retardos, es decir es más rápido. Es usado típicamente para el control de flujo de minerales. Función de Transferencia: $K_p sT_d$

- *Controlador proporcional integral derivativo (PID)*: este controlador es el más completo y complejo, tiene una respuesta más rápida y estable siempre que esté bien sintonizado. Resumiendo se puede decir que: El control proporcional actúa sobre el tamaño del error. El control integral rige el tiempo para corregir el error. El control derivativo le brinda la rapidez a la actuación. Función de Transferencia: $sT_d sT_i K_p + 1$

Moderno. Los métodos de control moderno brindan nuevas técnicas que permiten ya sea compensar el error y/o eliminarlo, las más comunes son las siguientes:

- *Control anticipatorio (Feedforward)*: este método permite al controlador analizar los datos

de entrada y de salida y mediante algoritmos matemáticos calculará la próxima salida probable, de modo tal que auto ajusta sus parámetros con la finalidad de adecuar se al cambio, y minimizar la diferencia de medidas. Se recomienda para procesos lentos. Su desventaja radica en que es necesario medir todas las variables perturbadoras, ya que no corrige las perturbaciones no medidas. Se puede mejorar este método agregando una retroalimentación a la salida, de modo tal que se deje que se produzca un error mínimo, el cual será detectado y corregido en la siguiente medición.

- *Compensadores adelanto atraso*: Este método permite realizar un control en el dominio de la frecuencia, en el cual se busca compensar la fase del sistema, agregando (adelanto) o quitando (atraso) fase, para lo cual se agrega nuevos componentes o nuevas funciones matemáticas al sistema. Se puede poner cuantos compensadores sea necesario a fin de llevar la respuesta del sistema a un valor deseado.

- *Compensador en Adelanto*.

- *Realimentación de estados*: este método permite ejercer una acción de control mediante el censado de cada uno de los estados (del modelo en espacio estado del sistema), atribuyéndola una ganancia a cada uno de los valores leídos, de este modo el lazo de control es cerrado por medio del compensador o controlador de estados y no por el sensor.

- *Sistemas de seguimiento*: este método también es conocido como Tracking, es un complemento del método anterior, puesto que mediante el control por realimentación de estados se puede llevar la variable controlada a un valor de cero (porque no se cuenta con una referencia), con este método se podrá llevar a la variable dada a un valor deseado, puesto que se incorpora una referencia en el sistema.

- *Feedback linealization*: debido a que los procesos reales no cuentan con modelos lineales que los representan, es necesario el uso de controladores no lineales. Este método es conocido como control con modelo de referencia..

Avanzado. Los métodos de control avanzado son aquellos que actúan en forma preventiva, de modo tal que en base a los datos tomados, actúan de modo tal que previenen la ocurrencia de error, por tanto el controlador está ajustando sus parámetros constantemente.

- *Control adaptativo*: es una variante del control anticipatorio, en donde la respuesta del controlador varía automáticamente basado en los cambios de las condiciones dentro del proceso, es decir, la respuesta del controlador será variable dependiendo del comportamiento actual del proceso. Para que se lleve a cabo esta adaptación se requiere de algoritmos matemáticos que simulen el proceso en base a los datos tomados en el instante mismo en que se realiza la acción, este resultado va a generar una señal compensadora que garantizará la confiabilidad del sistema.

- *Control óptimo*: el control óptimo busca la performance en la acción de control, tiene por objetivo buscar una o varias soluciones que cumplan con ciertas restricciones impuestas por el problema y que a la vez cumpla con una función objetivo (función de costo), la cual puede ser maximizar o minimizar dicha función.

- *Control robusto*: el control robusto es aquel que va a permitir mantener la acción de control pese a perturbaciones externas e internas. Puede existir perturbaciones externas como ruido y vibraciones propias del proceso; o perturbaciones internas como un mal modelamiento matemático, sistemas no lineales difíciles de linealizar, incertidumbre en el accionar o respuesta de la planta frente a estímulos, entre otros. El control robusto se resume a identificar y controlar la incertidumbre en los parámetros y en el comportamiento de una planta.

- *Control en tiempo real*: se define el control de sistemas en tiempo real, al control realizado en un intervalo de tiempo finito y constante, es decir que la información será censada con muestras intermitentes pero todas las veces con un mismo tiempo de muestreo. Características: Pueden realizar varias actividades en paralelo. Pueden ejecutar tareas en respuesta a señales externas. Deben funcionar en presencia de fallos o averías parciales, haciendo uso de elementos redundantes. Adquieren datos del exterior. Puede ser pasiva cuando utilizan interrupciones, o activa mediante el uso de tarjetas de entrada/salida de señales. Necesitan de un sistema operativo que les brinde: gestión eficiente de interrupciones, planificación de tareas y priorización de las mismas, acceso a puertos e interfaces, mecanismos de medición del tiempo, entre otros. El sistema operativo más empleado es el Linux.

- *Control difuso*: se basa en la lógica difusa, la cual a diferencia de la lógica binaria o booleana (verdadero/falso ó 1/0), asigna valores intermedios dentro de esta escala. Utiliza la experiencia del operador para generar una lógica de razonamiento para el controlador. No requiere del modelamiento matemático de la planta, puede representar modelos de sistemas lineales y no lineales mediante el uso de variables lingüísticas y una serie de condiciones o reglas previamente definidas.

- *Control neuronal*: hace uso de neuronas de inteligencia artificial. La neurona artificial estándar es un elemento de procesamiento que calcula una salida multiplicando su vector de entradas por un vector de pesos y este resultado es aplicado a una función de activación; un conjunto de neuronas conforman una red neuronal. Las Redes Neuronales son parte de la Inteligencia Artificial (AI) caracterizadas por su capacidad de aprendizaje, su velocidad mediante el procesamiento masivo en paralelo de datos y por la facilidad de modelado de sistemas y controladores no lineales.

Características: son dispositivos no-lineales; pueden aprender un mapeo; son adaptables; las respuestas están basadas en evidencia; usan contexto, es decir, a más información; la respuesta es más veloz y mejor; son tolerantes a fallas, o sea, que la falla es degradada; su diseño y análisis es uniforme; tienen analogía neuro-biológica; su procesamiento es masivamente, en paralelo, distribuido y realizado con operaciones individuales simples; aprenden por sí mismas; y tienen capacidad de generalizar.

Aplicaciones: representación de comportamientos de funciones lineales y no lineales; identificación de patrones o sistemas; sistemas de control Reconocimiento de imágenes; reconocimiento de caracteres; reconstrucción de datos; predicción y recomendación para la toma de decisiones; simulación de modelos económicos y financieros; clasificación de objetos;

y predicciones de clima.

Aprendizaje: el aprendizaje es la acción de mejorar el comportamiento mediante la observación de un error pasado con la finalidad de disminuir el error. Eso se produce modificando los pesos de la red neuronal.

Tipos: supervisado, no supervisado y por refuerzo

Algoritmos genéticos: este método simula la evolución natural de las especies propuesta por Charles Darwin, fue ideado por John Holland en 1970. La información va sufriendo cambios igual que lo harían las especies, es decir, se van adaptando al entorno, lo cual se lleva a cabo por medio de los procesos de selección natural, mezcla, y mutación. En cada ciclo (iteración) una parte del conjunto de hipótesis conocido como población actual, es reemplazado por una nueva población mediante las funciones evolutivas anteriores. Así sucesivamente en cada ciclo la población es evaluada en base a una función evolutiva, siendo conservados los datos más exactos, y siendo eliminados los datos que presentan error (selección natural). Para conservar el número de individuos (datos) estos son mezclados, lo cual genera nuevos individuos similares a sus procreadores. Finalmente cada cierto tiempo o dada cierta cantidad de individuos, algunos de los nuevos individuos son mutados aleatoriamente, pudiendo ser conservados o eliminados en la próxima iteración dependiendo de su utilidad dentro del sistema.

Sistemas expertos: estos sistemas tratan de emular la experiencia adquirida por uno o más seres humanos a lo largo del tiempo para realizar un trabajo. Este sistema tendrá en su memoria una base de datos con múltiples soluciones a un mismo problema, luego el sistema tendrá que escoger de entre esas soluciones a la que pueda aplicar se a fin de lograr los mejores resultados. El sistema se creaba dándose en las experiencias humanas, la elección de la estructura de control dependerá de las características del trabajo en donde se aplicará, además el sistema podrá ir aprendiendo con el tiempo y almacenar sus propias experiencias, existe mucha analogía entre los sistemas expertos y los sistemas neuro-fuzzy.

Estrategias de control. De acuerdo a Mendiburu (2003) las estrategias de control se tiene del tipo:

Cascada (Cascade). Consiste en incluir uno o más lazos de control interno dentro de otro externo, con el objetivo de anular perturbaciones, impidiendo que estas perturbaciones secundarias afecten al sistema principal. Básicamente el controlador externo se encarga de la variable principal, mientras que los controladores internos se encargan de las perturbaciones más frecuentes. Como regla general, a más interno es el lazo, la respuesta de este debe ser más rápido. Ventajas: Las perturbaciones más frecuentes son corregidas antes de afectar a la variable principal. Permite usar ganancias altas.

Relación (Ratio). Consiste en analizar y mantener una proporcionalidad entre dos o más elementos (actuadores) dentro de un proceso continuo. Por ejemplo se usa comúnmente cuando

tienen que ingresar dos líquidos a un tanque, y donde la cantidad del primer líquido debe ser el doble que la del segundo, además los líquidos deben entrar constantemente al tanque. Para controlar este tanque se hará uso de un sensor de flujo, un controlador y un actuador, por cada línea. Sin embargo si se aplica control por relación, se hará uso de dos sensores de flujo, un controlador, un actuador, y un control de relación, lo que significa un ahorro de instrumentos y un sistema mas sencillo de supervisar y reparar.

Rango dividido (Split range). Es aplicado a sistemas con una sola variable controlada y dos o más variables manipuladas, las cuales afectan de igual forma a la variable controlada. Requiere compartir la señal de salida del controlador con varios elementos actuadores.

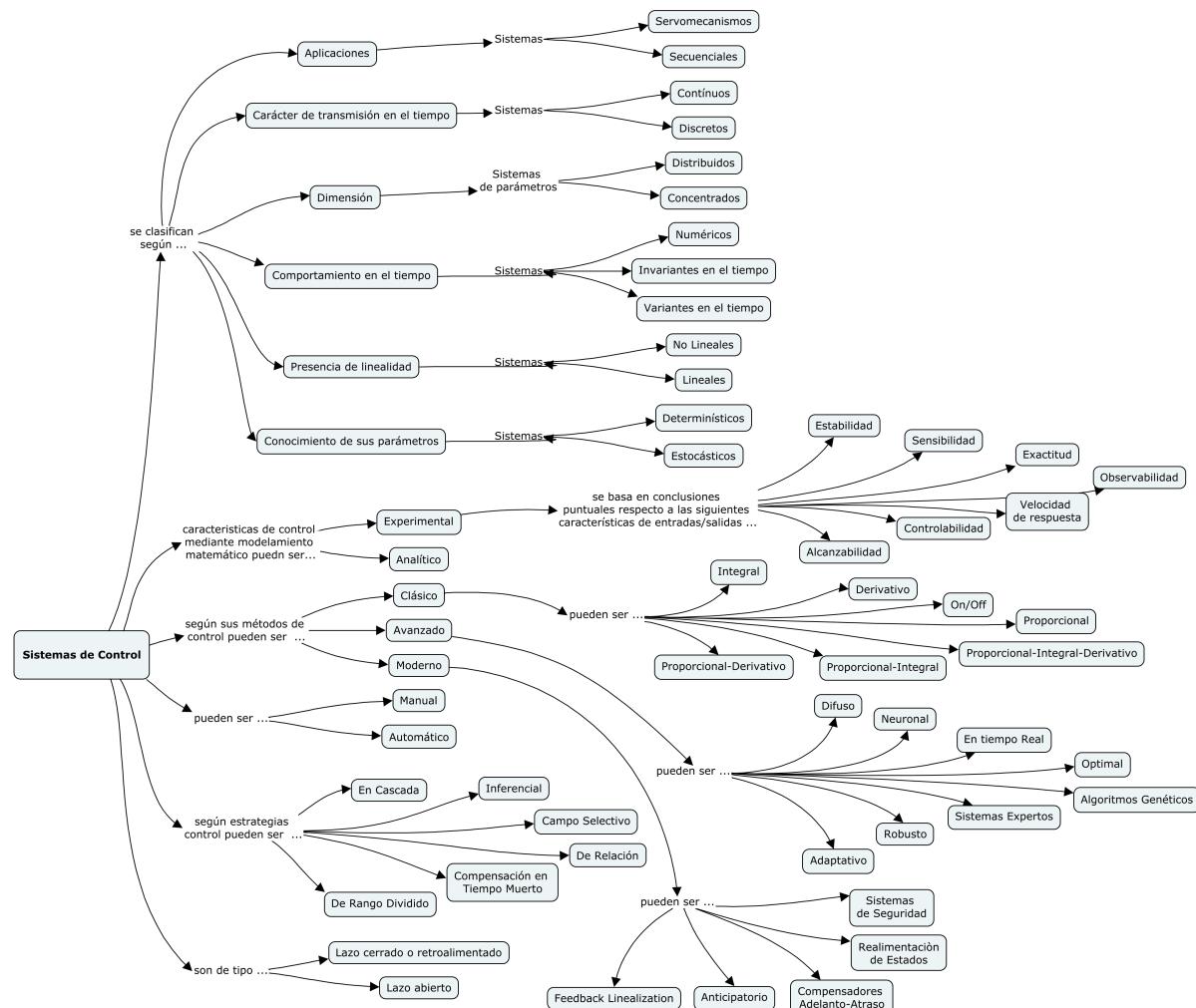
Selectivo (Override). Consiste en ejercer control sobre dos variables de un proceso, relacionas entre sí de tal modo que una u otra pueda ser controlada por la misma variable manipulada. La acción de control se logra conectando la salida de los controladores a un switch selector. Es aplicado en seguridad y protección de equipos y motores.

Inferencial. Consiste en efectuar la medición de la variable controlada a través de otra variable relacionada, considerada variable secundaria (pero dependiente de la principal). Los componentes de este sistema son los mismos que los de un sistema de control realimentado más una unidad de computo llamada estimador. Se aplica a procesos donde la obtención de información o la medición no se puede llevar a cabo por motivos de que no existe un elemento medidor para ese tipo de parámetros, o si existe es demasiado caro, o también porque no se puede medir constantemente el parámetro, lo que hace que se adquieran muy pocas muestras en un tiempo muy largo. Por ejemplo tenemos la medición del contenido de humedad en sólidos en operaciones de secado.

Compensación de tiempo muerto. El tiempo muerto es el intervalo de tiempo de respuesta desde que se ingresa una señal en la entrada a un componente o un sistema, y el comienzo de una señal de respuesta por la salida del sistema. El tiempo muerto presenta la principal dificultad en los diseños de sistemas de control estable.

Mapa conceptual de sistema de control.

Figura 9: Mapa conceptual sobre sistema de control. Fuente: Autor



Sistemas de automatización. Mendiburu (2003) expresa que en los sistemas automatizados la inteligencia del proceso está dado no por el ser humano, sino que es gobernada por una unidad de control. La tecnología usada ha adoptado diferentes formas desde automatismos mecánicos hasta los actuales automatismos con inteligencia artificial. La combinación de la inteligencia de los autómatas programables con los accionadores industriales, así como el desarrollo de captadores y accionadores cada día más especializados, permite que se automatice un mayor número de procesos. Los autómatas han evolucionado incorporando nuevas formas de procesar la información y ampliando sus funciones. Elementos de una instalación automatizada:

- **Máquinas:** son los equipos mecánicos que realizan los procesos, traslados, transformaciones, etc., de los productos o materia prima.
- **Accionadores:** son equipos acoplados a las máquinas, y que permiten realizar movimientos, calentamiento, ensamblaje, embalaje. Pueden ser:

Accionadores eléctricos: usan la energía eléctrica, son por ejemplo, electroválvulas, motores, resistencias, cabezas de soldadura, etc.

Accionadores neumáticos: usan la energía del aire comprimido, son por ejemplo, cilindros, válvulas, etc.

Accionadores hidráulicos: usan la energía de la presión del agua, se usan para controlar velocidades lentes pero precisas.

- *Pre-accionadores*: se usan para comandar y activar los accionadores. Por ejemplo, contactores, switchs, variadores de velocidad, distribuidores neumáticos, etc.

- *Captadores*: son los sensores y transmisores, encargados de captar las señales necesarias para conocer el estados del proceso, y luego enviarlas a la unidad de control.

- Interfaz hombre-máquina: permite la comunicación entre el operario y el proceso, puede ser una interfaz gráfica de computadora, pulsadores, teclados, visualizadores, etc.

- *Elementos de mando*: son los elementos de cálculo y control que gobiernan el proceso, se denominan autómatas, y conforman la unidad de control.

Los sistemas automatizados se conforman de dos partes:

Parte de mando: Es la estación central de controlo autómata . Es el elemento principal del sistema, encargado de la supervisión, manejo, corrección de errores, comunicación, etc.

Parte operativa: Es la parte que actúa directamente sobre la máquina, son los elementos que hacen que la máquina se mueva y realice las acciones. Son por ejemplo; los motores, cilindros, compresoras, bombas, relés, etc.

Autómatas programables. Según Mendiburu (2003) los autómatas son unidades de control, conformadas por equipos electrónicos, los cuales cuentan con un cableado interno independiente del proceso a controlar (hardware), y mediante algoritmos definen la estrategia y caminos a seguir para controlar un proceso (software). Estos equipos son conectados hacia los dispositivos de medida (sensores) y hacia los dispositivos encargados de realizar acciones (actuadores). Una vez que el autómata esta provisto de un algoritmo en su memoria, y que además cuenta con las conexiones respectivas hacia los equipos de planta, se convierte en el cerebro de la fábrica, teniendo a su cargo el desempeño y funcionamiento de toda la cadena productiva; los operarios deben supervisar que este se comporte tal como se había planificado y conforme a la lógica ideada por el programador.

Funciones del autómata y del sistema. Según Mendiburu (2003) el autómata debe realizar simultáneamente muchas funciones, siendo las principales: detección y lectura de las señales que envían los captadores, elaborar y enviar las acciones de mando y control al sistema, a través de los accionadores y pre-accionadores, mantener un diálogo con los operarios, informando el estado del proceso y detectando fallas, permite ser reprogramado con un nuevo algoritmo de supervisión y control. El conjunto de elementos que conforman el sistema automatizado deben de interactuar y desempeñar ciertas funciones que son de carácter general para todo el

sistema, estas son: establecer comunicaciones entre las diversas partes del sistema, así como la comunicación con otros sistemas, realizan tareas de supervisión y detección de fallas, deben controlar tanto procesos continuos, como procesos discretos y a través de un bus de campo o cable de comunicación, deben recoger la información de procesos remotos.

Estructura de un autómata. Según Mendiburu (2003) las unidades autómatas programables, internamente se ven constituidas por los siguientes dispositivos:

- *Fuente de alimentación:* brinda el suministro eléctrico a la unidad, permite la conversión de la energía alterna de la red eléctrica a energía continua requerida por los componentes electrónicos.

- *Tarjeta procesadora:* es el cerebro del autómata programable que interpreta las instrucciones que constituyen el programa grabado en la memoria, interpreta y deduce las operaciones a realizar, y acciones de control o supervisión a llevarse a cabo.

- *Tarjeta de memoria:* contiene los componentes electrónicos que permiten memorizar el programa, los datos de los sensores (señales de entrada) y los accionadores que deben realizar los actuadores (señales de salida).

- *Módulos de entrada/salida:* son tarjetas que permiten el conexión con dispositivos de entrada y/o salida tanto de tipo analógico como de tipo digital

La unidad autómata puede ver se complementada con unidades periféricas de entrada y salida, como es una consola de programación para digitar e ingresar los algoritmos, o una pantalla para una presentación gráfica del proceso y los parámetros del sistema.

Selección del autómata. Teniendo en cuenta a Mendiburu (2003) la selección del autómata que se va a emplear va a depender del tipo de proceso y de las funciones que se quiera realice el sistema. Hay que tener en cuenta los siguientes factores:

- *Factores Cuantitativos:* toma en consideración factores numéricos y de cantidad.

* Entradas y salidas: se debe determinar la cantidad de señales de entrada y salida que existan en el sistema y que merezcan conectarse al controlador, luego se debe determinar si las entradas/salidas son de tipo analógico o de tipo discreto. Conocida la cantidad, a este total se le debe agregar entre 10 y 20% adicional (reserva para futuras ampliaciones). Dependiendo de la ubicación de los elementos que van conectados al controlador, puede darse el caso que se encuentren muy lejanos por lo que se presentará problemas de atenuación y ruido en el cableado; frente a esto se puede optar por el control distribuido, es decir, colocar varios controladores en distintos puntos de las instalaciones, y cada quien manejará un determinado número de entradas/salidas.

* Memoria: se debe considerar la memoria del sistema y la memoria lógica. La cantidad de memoria del sistema esta directamente ligado al número de entradas y salidas y al tipo de estas, así tenemos que una entrada/salida digital ocupa 1 bit de información, mientras que una entrada /salida analógica ocupa 16 bits. La memoria lógica esta referida a la cantidad de información

que se debe de almacenar a raíz del algoritmo de control, cada instrucción va a sumar 1 ó 2 bytes, pero los comandos de mayor jerarquía (timers, contadores, sumadores, conversores, etc.) necesitarán más memoria. Se debe considerar el tamaño de las memorias, luego adicionarles un porcentaje de reserva, y ubicar la memoria comercial más acorde con las necesidades (1K, 2K, 4K, 8Kbytes, etc.)

* Alimentación: dependiendo de la cantidad de módulos de entrada/salida que se tenga que gestionar, el autómata requerirá mayor nivel de amperaje a un voltaje constante, por cuanto la fuente de alimentación debe estar planificada para soportar dicho requerimiento de corriente. Adicionalmente es recomendable contar con fuentes de reserva en caso de que la principal deje de operar.

* Periféricos: hay que considerar que el autómata puede conectarse a dispositivos externos, para lo cual debe contar con los puertos necesarios para la conexión. En el mercado existen autómatas a quienes se les puede conectar impresoras, monitores, unidades de disco, visualizadores y teclados alfanuméricos, unidades de cinta, etc.

- *Factores Cualitativos:* toma en consideración factores de calidad, calidad, desempeño, y modo de trabajo.

* Condiciones físicas y ambientales: el ambiente de trabajo en donde debe operar el autómata es determinante cuando se debe elegir la confiabilidad y robustez del equipo, puesto que un equipo de mayor calidad es más costoso, la planificación debe considerar no sobreestimar las condiciones del ambiente (polvo, humedad, temperatura) y requerir un equipo de mucha mayor robustez al realmente necesitado. En general los fabricantes realizan una serie de pruebas cuyos resultados se reflejan en las características técnicas de los equipos: rango de temperatura de trabajo y almacenaje, vibración soportada, nivel de interferencia, humedad, tipo de case, etc.

* Tipo de control: determinar el tipo de control a emplear es una función de los ingenieros de control, y dependerá de la complejidad del proceso, a sí como de la necesidad de contar con backups. Se puede optar por un control centralizado o uno distribuido, por un control PID o uno adaptativo, etc.

* Servicios adicionales: esta dado por las ventajas adicionales con que cuenta un equipo en relación a otro, como puede ser: el software de programación puede ser más amigable, más comprensible, con un entorno gráfico, ejemplos desarrollados, etc.; ciertos equipos pueden dar una mayor garantía, que cubre más situaciones de operación, o simplemente cubren por un mayor lapso de tiempo; el trato del suministrador también es importante a la hora de decidirse por un equipo u otro, además los suministradores pueden brindar cursos gratuitos de capacitación para el persona la cargo, asistencia técnica y mantenimiento permanente; disponibilidad en stock dentro del país, del producto así como de los componentes internos (repuestos, en caso sea necesario una reparación); etc.

* Compatibilidad: en algunos casos se preferirá equipos de tipo estándar, mientras que en otros casos será necesario equipos de tipo propietario. La elección del equipo en cuanto a su compatibilidad estará ligada a los demás equipos existentes en la planta. En algunos casos hay

quienes prefieren una marca porque le tienen confianza y ya les ha dado buenos resultado, en cambio hay otros que no quieren amarrarse con un solo suministrador y prefieren usar equipos compatibles.

Ciclos del programa. Teniendo en cuenta a Mendiburu (2003) el controlador o autómata programable, tiene almacenado en su memoria un algoritmo o programa de instrucciones, este algoritmo contiene un programa principal o parte fundamental, y subprogramas o partes secundarias, para las aplicaciones con autómatas programables el usuario programador necesita de una comunicación con la máquina para programar y depurar el programa, para acceder a los estados de planta y para forzar secuencias de mando sobre el sistema, para llevar a cabo estas funciones se requiere de unidades de programación, equipos y software, con interfaces sencillas para el usuario, y con canales y protocolos de conexión con el autómata estandarizados. El operador debe poder acceder a las funciones que necesita (programación, depuración, visualización, forzado, etc.) de forma rápida y flexible, según procedimientos que le facilitan la utilización del autómata en cualquier proceso industrial. Estos equipos auxiliares son conformados por unidades de programación, diagnóstico y test, destinadas a facilitar la edición y puesta a punto de programas para el autómata; Las unidades de programación utilizadas son en su mayoría computadoras personales.

La ejecución de un programa dentro de un controlador programable sigue un esquema cíclico totalmente diferente al que sigue un algoritmo diseñado para computadora.

Un ciclo de programa se compone de:

- *Operaciones de gestión del sistema:* que consiste en el tratamiento de la información, tratamiento de las peticiones y las llamadas efectuadas por el terminal de programación y el envío de mensajes al terminal.

- *Lectura del estado de las entradas:* consiste en la lectura de los registros de los módulos de entrada y el almacenamiento de estos datos en una memoria.

- *Ejecución del programa almacenado:* Empieza por la primera línea y sigue ejecutando línea por línea hasta la última. Durante esta ejecución no se considera una posible variación en el estado de las entradas, el estado de las entradas te toma del valor almacenado en la memoria. Una vez ejecutada la última línea se graban los resultados nuevamente en la memoria.

- *Escritura de datos de salida:* se efectúa una transferencia de información de la memoria a los módulos de salida. El último valor que tome la variable de salida será almacenado en la memoria.

La ejecución del programa principal y los sub-programas, puede realizarse de dos maneras:

- *Ejecución cíclica:* consiste en encadenar los ciclos del programa uno tras otro en forma continua y sin interrupciones. Después de actualizar las salidas, el sistema pasa a realizar el ciclo nuevamente y así sucesivamente. Es el método de ejecución por defecto.

- *Ejecución periódica:* consiste en ejecutar el programa durante cierto tiempo, o también durante un número definido de veces. Estos criterios pueden ser definidos por el operador.

Fundamentos de la electricidad.

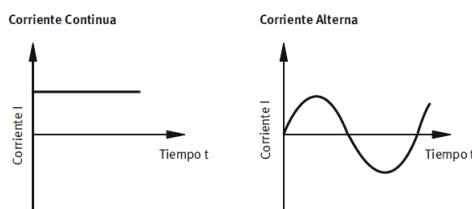
Corriente continua y corriente alterna. Según Ebel *et al.* (2007) una de las tecnologías más importantes en automatización es la electrotecnia, porque la mayoría de los sistemas técnicos necesitan energía eléctrica para funcionar y, además, para procesar las señales de entrada. Por ello, a continuación se ofrecen informaciones generales sobre los fundamentos esenciales de la electrotecnia.

Un circuito eléctrico sencillo está compuesto por una fuente de tensión, una unidad consumidora y de los cables para la conducción de la energía eléctrica. En cualquier circuito eléctrico se aplica una regla muy sencilla: «Desde la fuente hacia la unidad consumidora y de regreso». En términos físicos, en un circuito eléctrico los portadores de carga eléctrica negativa, es decir, los electrones, avanzan a través del conductor desde el polo negativo de la fuente de tensión hacia el polo positivo. Este movimiento de los portadores de carga se llama corriente eléctrica. Una corriente eléctrica únicamente puede fluir si el circuito eléctrico está cerrado.

Se puede diferenciar entre corriente continua y corriente alterna:

- Si la tensión en un circuito siempre actúa en un mismo sentido, la corriente siempre fluye en un mismo sentido. En ese caso, se trata de corriente continua, es decir, de un circuito de corriente continua.
- Tratándose de corriente alterna, es decir, de un circuito de corriente alterna, la tensión y la intensidad cambian su sentido y carga en una frecuencia determinada.

Figura 10: Corriente continua y corriente alterna en función del tiempo. Fuente: Ebel *et al.* (2007)



Sentido técnico del flujo de la corriente: antes de conocerse la existencia de los electrones, se determinó que la corriente eléctrica fluía de «positivo» a «negativo». En la práctica, esta definición sigue siendo válida en la actualidad. Se trata de la definición técnica del sentido del flujo de la corriente eléctrica.

Resistencia eléctrica y potencia eléctrica.

Conductor eléctrico. Según Ebel *et al.* (2007) bajo corriente eléctrica se entiende el movimiento rectificado de portadores de carga. Una corriente únicamente puede fluir en un material si éste contiene una cantidad suficiente de electrones libres. Los materiales que cumplen esta condición se llaman conductores eléctricos. Los metales cobre, aluminio y plata son

conductores especialmente buenos. En la técnica de control se utiliza principalmente el cobre como material conductor.

Resistencia eléctrica. Según Ebel *et al.* (2007) cualquier material, aunque sea buen conductor, ofrece una resistencia a la corriente eléctrica. Esta resistencia se produce porque los electrones libres chocan con los átomos del material conductor, por lo que se inhibe su movimiento. En el caso de los materiales conductores, la resistencia es menor. Los materiales que ofrecen una gran resistencia al flujo de la corriente eléctrica se llaman aislantes eléctricos. Para aislar los cables eléctricos se utilizan materiales que son mezclas de goma o de plástico.

Ley de Ohm. Según Ebel *et al.* (2007) la ley de Ohm describe la relación entre la tensión, la intensidad y la resistencia. Según esta ley, en un circuito eléctrico que tiene una resistencia determinada, la intensidad de la corriente cambia según cambia la tensión, es decir, si aumenta la tensión, también aumenta la intensidad y si baja la tensión, también baja la intensidad.

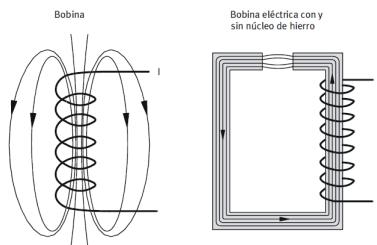
$V = R \times I$ donde V = Tensión [Unidad: Voltios (V)], R = Resistencia [Unidad: Ohmios (Ω)] e I = Intensidad [Unidad: Amperios (A)].

Potencia eléctrica. Según Ebel *et al.* (2007) en la mecánica, la potencia se define en función del trabajo. Cuanto más rápidamente se ejecuta el trabajo, tanto mayor debe ser la potencia. Por lo tanto, potencia significa: trabajo por unidad de tiempo, tratándose de una unidad consumidora incluida en un circuito eléctrico, la energía eléctrica se transforma en energía cinética (por ejemplo, movimiento giratorio de un motor eléctrico), en radiación de luz (por ejemplo, lámpara eléctrica) o en energía térmica (calefacción eléctrica, lámpara eléctrica). Cuanto más rápidamente se transforma la energía, tanto mayor es la potencia eléctrica. Por lo tanto, en este caso potencia significa lo siguiente: energía transformada por unidad de tiempo. La potencia aumenta en la medida en que aumentan la intensidad y la tensión. La potencia eléctrica de una unidad consumidora también se llama consumo eléctrico.

$P = V \times I$ donde P = Potencia [Unidad: Vatios (W)], V = Tensión [Unidad: Voltios (V)] e I = Intensidad [Unidad: Amperios (A)]

Funcionamiento de un electroimán. Según Ebel *et al.* (2007) alrededor de cualquier conductor por el que fluye corriente eléctrica se crea un campo magnético. Si se aumenta la intensidad, aumenta el campo magnético. Los campos magnéticos tienen un efecto atrayente para piezas de hierro, níquel o cobalto. Esta fuerza de atracción aumenta al aumentar el campo magnético. Las bobinas se miden en Henrios (H). El valor que tiene una bobina depende de: el número de espiras que tenga la bobina (a más vueltas mayor inductancia, o sea mayor valor en Henrios), el diámetro de las espiras (a mayor diámetro, mayor inductancia, o sea mayor valor en Henrios), la longitud del cable de que está hecha la bobina y el tipo de material de que está hecho el núcleo si es que lo tiene.

Figura 11: Bobina eléctrica sin y con núcleo de hierro y las correspondientes líneas de campo magnético. Fuente: Ebel *et al.* (2007)



Estructura de un electroimán. Según Ebel *et al.* (2007) un electroimán tiene la siguiente estructura: el conductor, por el que fluye la corriente eléctrica, se arrolla en forma de una bobina (bobina sin núcleo). El campo magnético aumenta debido a la superposición de las líneas de campo de todas las espiras de la bobina, se introduce un núcleo de hierro en la bobina. Si fluye corriente eléctrica, se magnetiza adicionalmente el hierro. Sin cambiar la intensidad, es posible obtener de esta manera un campo magnético mucho mayor que con una bobina sin núcleo. Aplicando ambas medidas, un electroimán atrae piezas ferríticas con gran fuerza, aunque la intensidad sea pequeña.

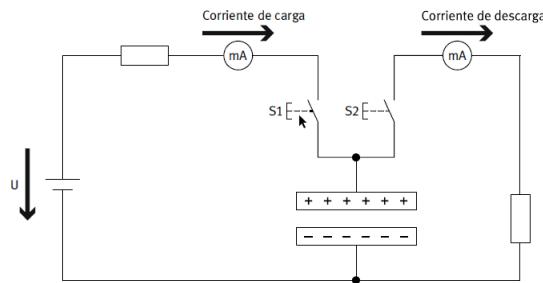
Aplicaciones de electroimanes. Según Ebel *et al.* (2007) en sistemas de control electroneumáticos, los electroimanes se utilizan principalmente para la commutación de válvulas, relés o contactores. Ejemplo de aplicación: válvula de vías de reposición por muelle, si fluye una corriente eléctrica a través de la bobina, se acciona el émbolo de la válvula y si se interrumpe el flujo de corriente, el muelle presiona sobre el émbolo de la válvula para que vuelva a su posición inicial.

Funcionamiento de un condensador eléctrico. Según Ebel *et al.* (2007) un condensador está compuesto por dos conductores (armaduras) separados por una capa aislante (dieléctrico). Si se conecta un condensador a una fuente de tensión continua, fluye brevemente una corriente de carga. Por ello, las dos armaduras se cargan eléctricamente.

Si, a continuación, se interrumpe la conexión con la fuente de tensión, la carga queda almacenada en el condensador. Cuanto mayor es la capacidad del condensador, tanto mayor es la cantidad de portadores de carga, siendo igual la tensión. La capacidad C es una magnitud importante para describir las características de un condensador. La capacidad expresa la relación entre la cantidad de portadores de carga Q y la tensión V conectada al condensador.

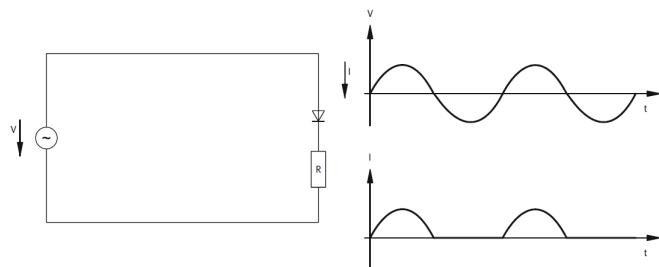
$C = \frac{Q}{V}$, la capacidad se expresa en «faradios» (F). Si se conecta una unidad consumidora al condensador cargado eléctricamente, se produce una compensación de carga. La corriente eléctrica fluye a través de la unidad consumidora hasta que el condensador está completamente descargado.

Figura 12: Funcionamiento de un condensador. Fuente: Ebel *et al.* (2007)



Funcionamiento de un diodo. Según Ebel *et al.* (2007) los diodos son semiconductores eléctricos cuya resistencia varía dependiendo del sentido de flujo de la corriente eléctrica. Si el diodo abre el paso en el sentido de flujo, la resistencia es mínima, de modo que la corriente eléctrica puede fluir casi sin resistencia. Si el diodo está cerrado en el sentido de flujo, la resistencia es muy alta, por lo que no fluye corriente eléctrica. Si se monta un diodo en un circuito eléctrico de corriente alterna, la corriente eléctrica únicamente puede fluir en un sentido. Ello significa que la corriente está rectificada. El efecto que un diodo tiene en la corriente eléctrica puede compararse al efecto que tiene una válvula de un neumático en el paso de aire: permite la entrada de aire al neumático, pero no permite que salga el aire.

Figura 13: Funcionamiento de un diodo. Fuente: Ebel *et al.* (2007)



Funcionamiento y estructura de interruptores y conmutadores. Según Ebel *et al.* (2007) para permitir o interrumpir el flujo de corriente en un circuito eléctrico, se utilizan interruptores o conmutadores. En principio, puede distinguirse entre interruptores tipo pulsador e interruptores con enclavamiento. Los pulsadores mantienen la posición de conmutación únicamente mientras se mantienen pulsados. Una aplicación típica de pulsadores es, por ejemplo, el timbre de una casa. Los interruptores con enclavamiento, por lo contrario, mantienen su posición de conmutación (ON/OFF). Estos interruptores mantienen su posición hasta que son accionados nuevamente. Una aplicación típica de pulsadores con enclavamiento son, por ejemplo, los interruptores de luz en una casa.

Otro criterio para clasificar y elegir interruptores es su estado de conmutación normal, es decir, cuando no están accionados.

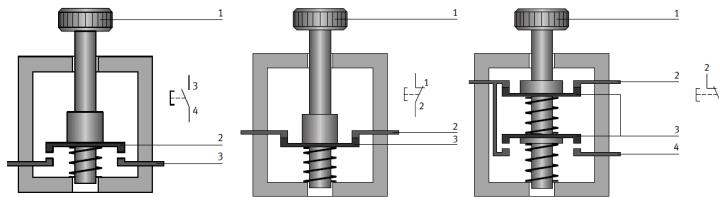
- *Contacto normalmente abierto:* en el caso de un contacto normalmente abierto, el circuito

de corriente está interrumpido mientras el interruptor se encuentra en su posición normal. Accionando el interruptor, se cierra el circuito eléctrico y se alimenta corriente eléctrica a la unidad consumidora. Soltándolo, el interruptor tipo pulsador recupera su posición normal por acción de un muelle, por lo que se interrumpe nuevamente el circuito eléctrico.

- *Contacto normalmente cerrado*: en el caso de un contacto normalmente cerrado, el circuito de corriente está cerrado por efecto de la fuerza del muelle mientras el interruptor se encuentra en su posición normal. Al accionar el pulsador, se interrumpe el circuito de corriente.

- *Comutador*: un conmutador combina en una sola unidad las funciones de un contacto normalmente cerrado y de un contacto normalmente abierto. Los conmutadores se utilizan para cerrar un circuito y abrir otro con una sola operación. Durante la operación de conmutación, los dos circuitos están interrumpidos durante unos breves instantes.

Figura 14: Contacto normalmente abierto, contacto normalmente cerrado y comutador: vista en corte y símbolo respectivamente. Fuente: Ebel *et al.* (2007)

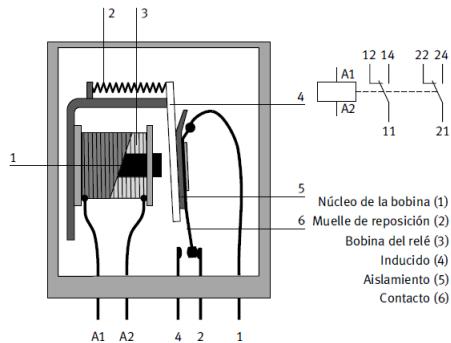


Relés y contactores.

Aplicaciones de relés. Según Ebel *et al.* (2007) en sistemas de control electroneumáticos se utilizan relés con los siguientes fines: multiplicar de señales, retardar y convertir señales, enlazar informaciones, separar el circuito de control del circuito principal y tratándose de sistemas de control puramente eléctricos, se utilizan adicionalmente para separar circuitos de corriente continua de circuitos de corriente alterna.

Estructura de un relé. Según Ebel *et al.* (2007) un relé es un interruptor accionado electromagnéticamente, en el que el circuito controlado y el circuito controlador están separados entre sí galvánicamente. Esencialmente está compuesto por una bobina con núcleo de hierro, un inducido como elemento de accionamiento mecánico (4), un muelle de recuperación (2) y los contactos de conmutación (6). Al conectar una tensión en la bobina del electroimán se produce un campo electromagnético. De esta manera, el inducido móvil es atraído por el núcleo de la bobina. El inducido actúa sobre los contactos del relé. Dependiendo del tipo de relé, los contactos se abren o cierran. Si se interrumpe el flujo de corriente a través de la bobina, el inducido recupera su posición inicial mediante la fuerza de un muelle. Con un relé se pueden activar uno o varios contactos. Además del tipo de relé antes descrito, existen otros tipos de interruptores o conmutadores accionados eléctricamente, como, por ejemplo, el relé de remanencia, el relé de temporización y el contactor.

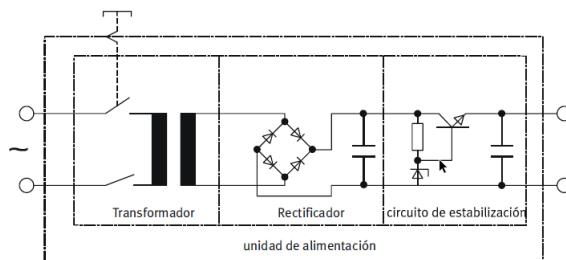
Figura 15: Relé: vista en corte y símbolo. Fuente: Ebel *et al.* (2007)



Funcionamiento y construcción de una unidad de alimentación. Según Ebel *et al.* (2007) las unidades de control reciben energía alimentada a través de la red eléctrica. Los grupos de la unidad de alimentación tienen las siguientes funciones:

- El transformador reduce la tensión de funcionamiento. La tensión de la red está conectada a la entrada del transformador (por ejemplo, tensión alterna de 230 V); en la salida, la tensión es menor (por ejemplo, tensión alterna de 24 V).
- El rectificador convierte la tensión alterna en tensión continua. El condensador que se encuentra en la salida del rectificador se utiliza para filtrar los picos de tensión.
- La regulación de la tensión en la salida de la unidad de alimentación es necesaria para que la tensión eléctrica sea constante, independientemente del flujo de la corriente.

Figura 16: Grupos de la unidad de un sistema de control electroneumático. Fuente: Ebel *et al.* (2007)

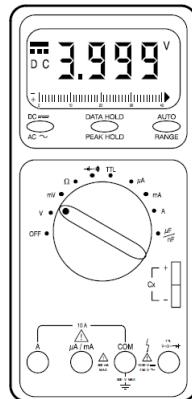


Mediciones en un circuito eléctrico. Según Ebel *et al.* (2007) medir significa comparar una magnitud desconocida (por ejemplo, la posición de un cilindro neumático) con una magnitud conocida (por ejemplo, la escala de una cinta de medición). Un aparato de medición (por ejemplo, una varilla de acero calibrada) permite llevar a cabo esa comparación. El resultado, es decir, el valor medido, se expresa mediante un número y una unidad (por ejemplo, 30,4 cm).

Las corrientes eléctricas, tensiones y resistencias suelen medirse con aparatos de medición múltiples. Estos aparatos pueden activar diversas modalidades de funcionamiento: tensión alterna/corriente alterna y tensión continua/corriente continua, o medición de intensidad, medición de tensión y medición de resistencias. La medición sólo puede ser correcta si se ajusta

el modo de funcionamiento correcto y si el aparato de medición se incluye correctamente en el circuito eléctrico.

Figura 17: Aparato de medición universal. Fuente: Ebel *et al.* (2007)

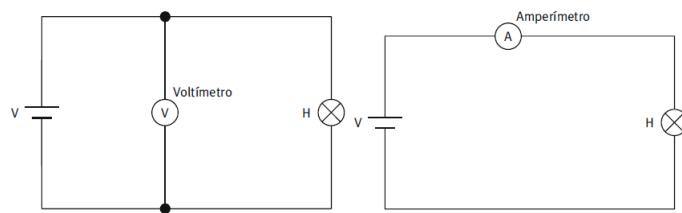


Según Ebel *et al.* (2007) al realizar mediciones en un circuito eléctrico, deberá procederse en el siguiente orden: desconectar la alimentación de tensión en el circuito eléctrico, ajustar la modalidad de funcionamiento en el aparato de medición universal (medición de intensidad o tensión, tensión continua o alterna, medición de resistencias), tratándose de aparatos de medición con escala y manecilla, comprobar el punto cero. En caso necesario, efectuar las correcciones del caso, al medir tensión continua/corriente continua, conectar el aparato de medición sin confundir los polos (borne «+» del aparato de medición al polo positivo de la fuente de tensión). Seleccionar el mayor margen de medición posible, conectar la alimentación de tensión, observar la reacción de la manecilla o la indicación en el visualizador y, paso a paso, seleccionar un margen de medición menor. Cuando se obtiene el movimiento máximo de la manecilla (con el margen de medición más pequeño posible), leer el resultado, si se trata de un aparato de medición con escala y manecilla, siempre leer perpendicularmente para evitar errores de lectura.

- *Medición de tensión:* según Ebel *et al.* (2007) al medir la tensión, el aparato de medición se conecta en paralelo en relación con la unidad consumidora. La caída de tensión a través de la unidad consumidora corresponde a la caída de tensión a través de la unidad de medición. Cualquier aparato utilizado para medir la tensión (voltímetro) tiene una resistencia interna. Para minimizar el error de medición, el flujo de corriente a través del aparato de medición debe ser mínimo. Es decir: la resistencia interna del voltímetro debe ser lo mayor posible.

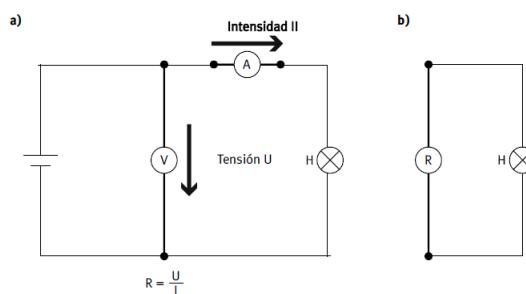
- *Medición de intensidad:* según Ebel *et al.* (2007) al medir la intensidad, el aparato de medición se conecta en serie en relación con la unidad consumidora. La corriente que fluye a través de la unidad consumidora también fluye a través del aparato de medición. Cualquier aparato utilizado para medir la intensidad (amperímetro) tiene una resistencia interna. Esta resistencia adicional disminuye el flujo de la corriente. Para minimizar el error de medición, el aparato de medición debe tener una resistencia interna muy pequeña.

Figura 18: Medición de tensión y de intensidad respectivamente. Fuente: Ebel *et al.* (2007)



- *Medición de resistencia:* según Ebel *et al.* (2007) la resistencia de una unidad consumidora incluida en un circuito de corriente continua puede medirse de modo directo o indirecto. En el caso de la medición indirecta, se mide la intensidad de la corriente que fluye a través de la unidad consumidora y la caída de tensión que se produce por efecto de dicha unidad consumidora. Ambas mediciones pueden realizar de modo consecutivo o simultáneamente. A continuación se calcula la resistencia aplicando la ley de Ohm. En el caso de la medición directa, la unidad consumidora se separa del resto del circuito eléctrico. Se activa la modalidad de funcionamiento «medición de resistencia» en el aparato de medición y, a continuación, se conectan los dos bornes a la unidad consumidora. La resistencia puede leerse directamente en el aparato de medición, si la unidad consumidora está dañada (por ejemplo, porque se quemó la bobina de una válvula), se obtiene un valor infinitamente alto o el valor cero (cortocircuito) al medir la resistencia.

Figura 19: Medición de resistencia. Fuente: Ebel *et al.* (2007)



Fundamentos de la electrónica. «La **electrónica** es la rama o campo de la ciencia y de la ingeniería que trata de la liberación, transporte, control, captura y conversión de energía de partículas subatómicas que tienen masa y carga (tales como los electrones) y que actúan en medios o materiales cuyas propiedades electromagnéticas son conocidas, por ejemplo, el vacío, los gases o los semiconductores» (Fink y Christiansen, 1992).

Según Alloza (2012) se denomina **componente electrónico** a un dispositivo que forma parte de un circuito realizando alguna función determinada. Estos elementos se suelen recubrir de algún material aislante (cerámico, plástico) dejando sus patillas o terminales fuera para permitir la conexión al circuito, es lo que se llama encapsulado, están construidos en su mayoría a partir de materiales semiconductores. Para observar símbolos electrónicos de los componentes electrónicos y ejemplos físicos ver el Anexo B.

Clasificación de los componentes electrónicos. De acuerdo a Malvino y Bates (2007) se tiene que todos los componentes electrónicos están clasificados dentro de dos grandes grupos, componentes pasivos y componentes activos; y estos a su vez pueden dividirse en otros grupos, dependiendo de sus características de funcionamiento.

Componentes pasivos. De acuerdo a Talca (2014) son aquellos que no pueden contribuir con la ganancia de energía o amplificación para un circuito o sistema eléctrico. Estos no tienen acción de control y no necesitan alguna otra entrada más que una señal para ejecutar su función. A este grupo pertenecen las resistencias, los condensadores, las bobinas, los conectores, los interruptores y los condensadores. Estos se pueden dividir en componentes pasivos lineales y componentes pasivos electromecánicos.

- *Componentes pasivos lineales:* son llamados así porque se comportan linealmente con la corriente o el voltaje, es decir, si aumenta o disminuye un voltaje, la corriente también aumenta en la misma proporción y viceversa. A este grupo pertenecen las resistencias, los condensadores y las bobinas.

- *Componentes pasivos electromecánicos:* son componentes pasivos que ejecutan funciones eléctricas simples a partir de movimientos mecánicos externos o internos. A este grupo también pertenecen los dispositivos que tienen funciones de soporte mecánico y de interconexión eléctrica. Podemos contar entre estos a los conductores, los interruptores, los conectores y los circuitos impresos, entre otros. Los principales componentes electromecánicos utilizados en los sistemas y circuitos electrónicos, haciendo énfasis en los interruptores y conectores. Estos componentes son importantes, ya que permiten la interconexión interna y externa de los circuitos y sistemas tanto eléctricos como electrónicos. Aunque su función es simple, es clave para su correcto funcionamiento aunque algunas veces no se les da la importancia que merecen.

Componentes activos. De acuerdo a Talca (2014) son aquellos que tienen la capacidad de controlar voltajes o corrientes y que pueden crear una acción de amplificación o de conmutación, esta es el intercambio de una señal entre dos estados en el circuito al que pertenecen. Entre ellos tenemos al diodo, los transistores, los tiristores y los circuitos integrados, entre otros. Estos se pueden dividir en componentes activos semiconductores y transductores. Los diodos no se consideran un verdadero componente activo ya que no produce amplificación. Sin embargo, están más relacionados con estos por su naturaleza semiconductor. Los semiconductores son en la electrónica en general y dentro de los componentes activos los más importantes, están basados en la propiedad que tienen ciertos materiales como el silicio (Si), selenio (Se) y germanio (Ge), por ejemplo, de comportarse como conductores o aislantes, bajo determinadas condiciones o estímulos externos. Esta propiedad se utiliza para rectificar corriente alterna, detectar señales de radio, amplificar señales de corriente eléctrica, funcionar como interruptores o compuertas utilizadas en electrónica digital, etc. Son llamados también dispositivos de estado sólido y son los verdaderos responsables de la revolución electrónica moderna. Entre los más empleados

tenemos el diodo, el transistor, el tiristor y el circuito integrado.

En los semiconductores hay dos tipos de portadores de corriente eléctrica: los electrones (carga negativa) y los huecos (carga positiva), a los materiales semiconductores puros se les conoce como semiconductores intrínsecos, y los semiconductores extrínsecos son materiales semiconductores puros contaminados con impurezas en mínimas proporciones (una partícula entre un millón). A este proceso de contaminación se le denomina dopaje.

Según el tipo de dopaje que se le realice al material existen dos tipos:

- *Tipo N*: en este caso se contamina el material con átomos de valencia 5, como son Fósforo (P), Arsénico (As) o Antimonio (Sb). Al introducirlos, fuerza al quinto electrón de este átomo a vagar por el material semiconductor, pues no encuentra un lugar estable en el que situarse. Al conjunto de estos electrones se les llama electrones mayoritarios. Al material tipo N se le denomina también donador de electrones.

- *Tipo P*: en este caso se contamina el material semiconductor con átomos de valencia 3, como son Boro (B), Galio (Ga) o Indio (In). Si se introduce este átomo en el material, quedando un hueco donde debería ir un electrón. Este hueco se mueve fácilmente por la estructura como si fuese un portador de carga positiva. En este caso, los huecos son portadores mayoritarios. Al material tipo P se le denomina donador de huecos (o aceptador de electrones).

Transductores: son aquellos componentes activos que convierten señales eléctricas en otras formas de energía y permiten que los sistemas electrónicos puedan interactuar con el mundo externo. A ellos pertenecen las pilas y las baterías, los micrófonos, los parlantes, las lámparas, los motores, etc.

Resistencias. De acuerdo a Talca (2014) son los componentes más comunes en los circuitos electrónicos y los de mas bajo costo. Se fabrican aprovechando la propiedad que tienen todos los materiales de ofrecer cierto grado de oposición al paso de la corriente y se emplean para controlar el paso de ella en los circuitos electrónicos. Estos dispositivos diseñados para este propósito se les conoce como resistencias (resistor), y su unidad es el OHM (Ω).

Según Miguel (2014) en los circuitos electrónicos, las resistencias cumplen un papel mucho más especial: permiten distribuir adecuadamente la tensión y la corriente eléctrica en los diferentes puntos del circuito. Para realizar esta correcta distribución se basan, en todo momento, en la ley de Ohm.

Tolerancia de una resistencia. Según Miguel (2014) en la fabricación de una resistencia con un valor exacto es muy difícil, por ello la tolerancia indica los valores máximos y mínimos entre los que estará comprendida la resistencia. Estos valores se expresan con un porcentaje del valor en ohmios asignado teóricamente. Ejemplo: +/- 0.5% para resistencias de alta precisión y la más utilizada es de +/- 10%.

Código de colores para resistencias. De acuerdo a Miguel (2014) la forma de inscribir el valor de una resistencia para que sea fácilmente identificable a simple vista, es la de utilizar una serie de anillos de colores pintados sobre la superficie del cuerpo de la resistencia, que, mediante un código, permite cubrir toda la gama de valores de resistencias existentes en el mercado.

Tabla 2: Código de colores para resistencias. Fuente: Miguel (2014)

Color	A 1 ^a cifra	B 2 ^a cifra	C Multiplicador	Tolerancia
Negro	0	0	× 1	
Marrón	1	1	× 10	± 1 %
Rojo	2	2	× 100	± 2 %
Naranja	3	3	× 1000	
Amarillo	4	4	× 10000	
Verde	5	5	× 100000	± 0,5 %
Azul	6	6	× 1000000	
Violeta	7	7	× 10000000	± 0,1 %
Gris	8	8	× 100000000	
Blanco	9	9	× 1000000000	
Oro	—	—	× 0,1	± 5 %
Plata	—	—	× 0,01	± 10 %
Sin color	—	—		± 20 %

Observación: Para la identificación de resistencias para montaje superficial (SMD) se utiliza un código alfanumérico de tres o cuatro dígitos.

Tabla 3: Ejemplos de resistencias SMD con tres dígitos. Fuente: Miguel (2014)

152	1ra. Cifra = 1er. número	Equivalente a 1.500 Ω o 1.5 kΩ
	2da. Cifra = 2do. número	
	3ra. Cifra = Multiplicador	
1R8	1ra. Cifra = 1er. número	Equivalente a 1.8 Ω
	La «R» indica la coma decimal	
	3ra. Cifra = 2do. número	
R33	La «R» indica «0»	Equivalente a 0,33 Ω
	2da. Cifra = 2do. número	
	3ra. Cifra = 2do. número	

Potencia de disipación de una resistencia. De acuerdo a Miguel (2014) la misión de una resistencia en un circuito electrónico no es precisamente la de calentarse, pero resulta inevitable que se produzca este fenómeno. La potencia depende de los valores de tensión e intensidad a que éste se someterá, es decir, cuanto mayor sea el tamaño físico de la resistencia, mejor podrá

evacuar o disipar el calor que produce. En el mercado existen resistencias que van desde 1/8 de vatios hasta más de 100 W.

Clasificación de las resistencias. Teniendo en cuenta a Miguel (2014) se mencionan la clasificación de las resistencias.

- *Resistencias fijas:* son aquellas en las que el valor en ohmios que posee es fijo y se define al fabricarlas.

Tabla 4: Tipos resistencias fijas. Fuente: Miguel (2014)

Resistencia Fija	Características
Aglomeradas	Están constituidas por una mezcla de grafito (o carbón), y un material aislante (resina, talco, etc.), en las proporciones adecuadas para obtener una determinada gama de valores. En los extremos del cilindro se colocan Por último se recubre el conjunto por una resina o se plastifica y se pintan los colores que indicarán el valor de la resistencia. Su valor cambia en exceso con la temperatura, por lo que son poco empleadas.
Película de carbón	Son las más usadas para pequeñas potencias. Consisten en un cilindro aislado en el que se deposita una delgada película de carbón con dos casquillos metálicos en los extremos. El valor óhmico de la resistencia, se practican unos surcos en espiral, se fabrican resistencias de muchos valores y de buena precisión. sobre este conjunto se deposita la capa de esmalte y se pintan los anillos de colores.
Película metálica	Éstas resistencias son básicamente iguales que las anteriores, con la diferencia de que utilizan una película de una aleación metálica que las hace muy estables con la temperatura. Con ellas se consiguen unas tolerancias muy bajas.
Bobinadas	están fabricadas a base de bobinar hilo resistivo (generalmente una aleación de Ni - Cr - Al) sobre un cilindro aislante hasta obtener el valor óhmico deseado. Se utilizan para grandes potencias, por lo que el recubrimiento exterior es de porcelana o esteatita. La tolerancia habitual es del % y son capaces de disipar potencias por encima de los 100 W.

- *Resistencias variables:* son resistencias a las que se les puede modificar su valor óhmico desde cero hasta su valor nominal. Para ello se les ha añadido un tercer terminal unido a un contacto móvil que puede desplazarse sobre el elemento resistivo proporcionando variaciones en el valor de la resistencia. Este tercer terminal puede tener un desplazamiento angular (giratorio) o longitudinal (deslizante). Según su función en el circuito estas resistencias se denominan:

- *Potenciómetros:* se aplican en circuitos donde la variación de resistencia la efectúa el usuario desde el exterior (controles de audio, vídeo, etc.).

- *Trimmers, o resistencias ajustables:* se diferencian de las anteriores en que su ajuste es definitivo en el circuito donde van aplicadas. Su acceso está limitado al personal técnico (controles de ganancia, polarización, etc.).

- *Reostatos:* son resistencias variables en las que uno de sus terminales extremos está eléctricamente anulado. Tanto en un potenciómetro como un trimmer, al dejar unos de sus terminales extremos al aire, su comportamiento será el de un reostato, aunque estos están diseñados para soportar grandes corrientes.

- *Resistencias dependientes:* son resistencias en que su valor óhmico se modifica bajo la acción de una variable física, como la temperatura, la luz, la tensión, la presión, la tracción mecánica, etc.

* Dependientes de la temperatura: son las de tipo NTC (Coeficiente Negativo de Temperatura) tiene la particularidad de disminuir la resistencia interna al aumentar su temperatura. También se llaman termistores. Pueden tener muchas aplicaciones entre las que podríamos destacar: la medida de temperatura en motores y máquinas, termostatos, alarmas contra calentamientos y de tipo PTC (Coeficiente Positivo de Temperatura) para compensación de circuitos eléctricos. La resistencia de tipo PTC aumenta la resistencia interna al aumentar la temperatura. Suelen utilizarse para protección de circuitos electrónicos.

* Dependientes de la luz (LDR): ciertos materiales como el Selenio varían sus propiedades conductoras cuando varía la intensidad de luz que incide sobre ellos. Este efecto se denomina fotoconductividad. Si construimos un circuito eléctrico formado por una pila, un amperímetro y un trozo de Selenio y hacemos incidir un fuerte rayo de luz sobre el Selenio, veremos que el amperímetro marca mayor paso de corriente. Las resistencias LDR, también llamadas fotoresistencias, tienen aplicaciones entre las que destacan puertas automáticas de ascensores, control del alumbrado público, alarmas, máquinas detectoras de luz (visión artificial), etc.

* Dependientes de la tensión (VDR): también llamadas varistores, este tipo de resistencia disminuye el valor óhmico al aumentar el voltaje eléctrico entre sus extremos.

* Dependientes de la inducción magnética (MDR): también llamados magnetoresistores, son utilizados como elementos sensibles o detectores de campos magnéticos. Ejemplo, la cabeza lectora del disco duro de un computador.

* Dependientes de las deformaciones y tensiones mecánicas: son llamadas también galgas extensiométricas, su resistencia varía de acuerdo a la deformación física a la que son sometidas, es decir, que al ser deformado aumenta su resistencia, puesto que los conductores se vuelven más largos y finos.

Condensadores. Según Miguel (2014) su característica principal es que tienen la capacidad de almacenar energía eléctrica en forma temporal. Los capacitores o condensadores, están formados básicamente por dos placas metálicas conductoras separadas por un material aislante llamado dieléctrico , este dieléctrico puede ser aire, mica, papel,cerámica, etc.La unidad de medida fundamental de los condensadores es el farad o faradio (f), denominada así en honor al físico francés Michael Faraday (1791 – 1867), descubridor de efectos magnéticos de las corrientes eléctricas. En la práctica el faradio es una unidad demasiado grande para la mayoría de las situaciones reales,por esta razón se utilizan unidades derivadas mas pequeñas como el microfaradio (μf) y el picofaradio (pf), equivalentes respectivamente a la millonésima (1×10^{-6}) y la billonésima (1×10^{-12}) parte de un faradio.

Aplicaciones de los condensadores. Descarga rápida, como un flash, en donde el condensador se tiene que descargar a gran velocidad para generar la luz necesaria (algo que hace muy fácilmente cuando se le conecta en paralelo un medio de baja resistencia).

Como Filtro, un condensador de gran valor se utiliza para eliminar el "rizado" que se genera en el proceso de conversión de corriente alterna en corriente continua.

Para aislar etapas o áreas de un circuito: un condensador se comporta (idealmente) como un cortocircuito para la señal alterna y como un circuito abierto para señales de corriente continua, etc.

Controlando el tiempo de carga y descarga de un condensador se pueden construir temporizadores. Para ello, hay que colocar una resistencia en serie con el condensador. El tiempo de carga y descarga de un condensador viene dado por la expresión: $t = 5xRxC$

Tipos de condensadores. Los condensadores, al igual que las resistencias pueden ser fijos o variables.

Tabla 5: Tipos condensadores fijos. Fuente: Miguel (2014)

Condensadores Fijos	Características
Electrolíticos	Tienen el dieléctrico formado por papel impregnado en electrolítico. Siempre tienen polaridad, y una capacidad superior a 1 μF .
Cerámicos	Se utilizan exclusivamente en microelectrónica, ya que sus valores y tamaños no son suficientes como para proporcionar las características que necesitaría el arranque de un motor, o el filtrado de una fuente de alimentación. Son sumamente baratos y suponen una opción de la que no se puede prescindir en muchos casos dadas sus características.
De plástico	Los condensadores de polímeros son muy utilizados, dado que entre sus características más importantes se encuentran una gran resistencia de aislamiento que le permite conservar la carga por largos períodos de tiempo, un volumen reducido y un excelente comportamiento frente a la humedad y a las variaciones de temperatura. Tienen además la propiedad de autoregeneración permite que en caso de que un exceso de tensión los perfore, el metal se vaporiza en una pequeña zona rodeando la perforación evitando el cortocircuito, lo que le permite seguir funcionando. Los materiales más utilizados son: poliestireno (styroflex), poliéster (mylar), policarbonato (Macrofol) y politetrafluoretileno (conocido como teflón). Se fabrican en forma de bobinas o multicapas. En algunos países o publicaciones se los conoce como MK. Se fabrican con capacidades desde 1nF a 100uF y tensiones desde 25V a 4000V. Se les distingue por sus características colores vivos, generalmente rojo, amarillo o azul.

Condensadores variables. De acuerdo Talca (2014) permite obtener valores de capacidad comprendidos entre un valor máximo y un valor mínimo preestablecido por el fabricante. Son accionados por medios mecánicos y dependiendo de si son diseñados para variar constantemente su valor o para ser ajustados a un valor determinado, pueden clasificarse como condensadores variables y condensadores de ajuste. En los condensadores variables sus láminas metálicas son móviles. La mitad de ellas están fijas y la otra mitad pueden accionarse mediante un eje, y hacer que entren en las ranuras que separan a las primeras variando así su superficie enfrentada en las placas. Su capacidad puede variar de los 5pf y los 500pf típicamente. Los dieléctricos empleados en los condensadores ajustables pueden ser: mica, vidrio, aireocerámica. Se basan también en la variación de superficie enfrentada entre las placas o la distancia que exista entre ellas. Son usados para compensar o ajustar pequeñas diferencias de calibración en los equipos y sistemas electrónicos. Son conocidos como trimmers y su capacidad es de muy pocos pico faradios.

Codificación de los condensadores. - Por código de colores.

Tabla 6: Código de colores para condensadores. Fuente: Miguel (2014)

Color	Banda 1	Banda 2	Multiplicador	Tolerancia (<10 pF)	Tolerancia (>10 pF)	Tensión máxima
Negro	-	0	1	±2 pF	±20 %	
Marrón	1	1	10	±0,1 pF	±1 %	100 V
Rojo	2	2	100	±0,25 pF	±2 %	250 V
Naranja	3	3	1.000			
Amarillo	4	4	10.000			400 V
Verde	5	5	100.000	±0,5 pF	±5 %	
Azul	6	6	1.000.000			630 V
Violeta	7	7				
Gris	8	8				
Blanco	9	9		±1 pF	±10 %	

- Por codificación con letras y números: otro sistema de inscripción del valor de los condensadores sobre su cuerpo. En lugar de utilizar bandas de colores se recurre a la escritura de diferentes códigos mediante letras y números impresos. Las letras que aparecen hacen referencia a la tolerancia: M = +/- 20%, K= +/- 10% y J= +/- 5%. Detrás de estas letras figura la tensión de trabajo y delante de las mismas el valor de la capacidad indicado con cifras. En ausencia de unidad se toma el microfaradio (μF) o bien se recurre al empleo del prefijo "n" para referirse al nanofaradio (nF). Los decimales se marcan con una coma (punto) decimal o poniendo la n en la posición que ocuparía.

- Por codificación 101: El código 101 se utiliza en los condensadores cerámicos como alternativa al código de colores. Con este sistema se imprimen 3 cifras, las dos primeras son las significativas y la tercera indica el número de ceros que se deben añadir a las precedentes. El resultado se expresa siempre en picofaradios (pF).

Bobinas. De acuerdo a Vassallo (2000) una bobina (también llamado inductor) es un operador o componente eléctrico formado por un conductor arrollado de forma cilíndrica. El arrollamiento está formado por varias capas de hilo de cobre electrolítico aislado con esmalte. La misión de una bobina es almacenar energía eléctrica en forma magnética para cederla en un momento determinado. La bobina viene definida por su autoinducción L, la cual depende del número de espiras que forman el arrollamiento (N), del flujo magnético que la atraviesa (Φ) y de la intensidad de corriente que la recorre (I), según la expresión: $L = N^2 \Phi / I$; la unidad de autoinducción es el henrio (H), pero, como se trata de una unidad muy grande, en electrónica se suelen emplear algunos submúltiplos, como el milihenrio (mH) y el microhenrio (μH).

$$1 \text{ mH} = 10^{-3} \text{ H}$$

$$1 \text{ } \mu\text{H} = 10^{-6} \text{ H}$$

Tipos de bobinas. 1. Fijas

- *Con núcleo de aire*: el conductor se arrolla sobre un soporte hueco y posteriormente se retira este quedando con un aspecto parecido al de un muelle. Se utiliza en frecuencias elevadas. Una variante de la bobina anterior se denomina solenoide y difiere en el aislamiento de las espiras y la presencia de un soporte que no necesariamente tiene que ser cilíndrico. Se utiliza cuando se precisan muchas espiras. Estas bobinas pueden tener tomas intermedias, en este caso se pueden considerar como 2 o más bobinas arrolladas sobre un mismo soporte y conectadas en serie. Igualmente se utilizan para frecuencias elevadas.

- *Con núcleo sólido*: poseen valores de inductancia más altos que los anteriores debido a su nivel elevado de permeabilidad magnética. El núcleo suele ser de un material ferromagnético. Los más usados son la ferrita y el ferroxcube. Cuando se manejan potencias considerables y las frecuencias que se desean eliminar son bajas se utilizan núcleos parecidos a los de los transformadores (en fuentes de alimentación sobre todo). Así nos encontraremos con las configuraciones propias de estos últimos. Las secciones de los núcleos pueden tener forma de EI, M, UI y L.

Las bobinas de nido de abeja se utilizan en los circuitos sintonizadores de aparatos de radio en las gamas de onda media y larga. Gracias a la forma del bobinado se consiguen altos valores inductivos en un volumen mínimo.

Las bobinas de núcleo toroidal se caracterizan por que el flujo generado no se dispersa hacia el exterior ya que por su forma se crea un flujo magnético cerrado, dotándolas de un gran rendimiento y precisión. La bobinas de ferrita arrolladas sobre núcleo de ferrita, normalmente cilíndricos, con aplicaciones en radio es muy interesante desde el punto de vista práctico ya que, permite emplear el conjunto como antena colocándola directamente en el receptor. Las bobinas grabadas sobre el cobre, en un circuito impreso tienen la ventaja de su mínimo coste pero son difícilmente ajustables mediante núcleo.

2. Variables

También se fabrican bobinas ajustables. Normalmente la variación de inductancia se produce por desplazamiento del núcleo. Las bobinas blindadas pueden ser variables o fijas, consisten encerrar la bobina dentro de una cubierta metálica cilíndrica o cuadrada, cuya misión es limitar el flujo electromagnético creado por la propia bobina y que puede afectar negativamente a los componentes cercanos a la misma.

Aplicaciones de las bobinas

Las principales aplicaciones de las bobinas son: los relés y los transformadores.

- *Relé*: es un elemento que se fundamenta en las propiedades del magnetismo para formar imanes no permanentes. Si introduces un trozo de hierro dulce en el interior de la bobina, cada vez que circule corriente eléctrica por ella se transformará en un imán. Cuando circule por la bobina, el hierro atraerá una pieza metálica que forma parte del conjunto. Esta pieza podrá bascular por uno de los extremos, de tal manera que el otro quedará libre y podrá cerrar o abrir un circuito. Cuando la corriente deje de circular, un muelle hará que esta pieza metálica vuelva a su estado inicial. De esta forma, según circule o no corriente, podremos abrir o cerrar circuitos

eléctricos.

- El transformador: es un componente basado en la disposición de dos bobinas acopladas magnéticamente con un núcleo de material ferromagnético, constituido por placas de ferrita (hierro). La bobina por donde entra la corriente recibe el nombre de primario y por donde sale de secundario. Debido a este acoplamiento, la señal de entrada en el transformador sufre variaciones que son recogidas a la salida. Se pueden utilizar transformadores para elevar su amplitud o disminuirla o bien para adaptar entre si otros componentes.

Conductores eléctricos. Teniendo en cuenta a Talca (2014) éstos pertenecen a la clasificación de los componentes electromecánicos, son dispositivos pasivos que efectúan funciones eléctricas simples, en este caso la interconexión entre componentes de un circuito eléctrico o electrónico y transportan señales de voltaje y corriente de un punto a otro, sin pérdidas apreciables, donde su característica más importante es su baja resistencia. Los mejores conductores son los metales siendo el mas utilizado el cobre, para aplicaciones especiales se usa el oro (procesadores) la plata (conectores), y el aluminio (transmisión de alta tensión).

Clasificación de los conductores eléctricos. - Alambre: constituidos por un solo hilo metálico de forma cilíndrica llamado alma, pueden estar desnudos o revestidos por una cubierta aislante. La parte conductora es metálica, generalmente de cobre recocido, aunque en algunos casos se usa oro y plata en electrónica y el aluminio para el transporte de energía eléctrica a grandes distancias, por ser más liviano. El fin del revestimiento es aislar eléctricamente el elemento conductor y protegerlo contra la humedad, la oxidación, el calor y otras condiciones extremas, y la más importante, evitar que haga contacto con otros conductores. El aislamiento también permite identificar las funciones de los conductores por su color y grabar sobre él los códigos de aislamiento, el número de conductor, el máximo voltaje de trabajo, máximo punto de temperatura de operación y otros datos que puedan interesar al usuario.

- Cable: construido por un conjunto de alambres no aislados entre sí. Puede estar revestido por uno o más capas aislantes.

Se dividen en 2 grupos:

* Trenzados: formado por varios alambres gruesos enrollados en forma de hélice o trenza. Se usan para hacer conexiones permanentes, no sometidas a flexiones, pero sí a trabajos pesados. De uso industrial, para tareas donde se maneja mucha potencia.

* Cable multifilar: formado por varios alambres muy delgados prácticamente hilos o filamentos. En electrónica uno de los más usados es el multifilar sencillo con diferentes espesores y colores en el aislamiento.

* Multiconductores: formado por varios alambres individuales sólidos o multipares aislados entre si y envueltos en una chaqueta común. (cable UTP)

* Cable coaxial o blindado: formado por un conductor central o alambre multifilar rodeado por una cubierta de polietileno gruesa, llamado dieléctrico, sobre la cual se encuentra un segundo

conductor trenzado en forma de malla, llamado blindaje. Se usa en comunicaciones, vídeo y aplicaciones de alta frecuencia.

* Cables ribbon o tipo cinta: en la actualidad muy usados; constituidos por varios conductores individuales dispuestos en forma de cinta unidos por sus aislamientos. Se usan en computadores, TV, equipos de sonido, etc., en sus extremos lleva un conector especial, dependiendo de la necesidad.

Interruptores. Según Talca (2014) son elementos que permiten, interrumpen o dirigen el paso de la señal eléctrica por un circuito. Aunque la función de estos componentes es bastante simple, en el momento de elegirlos debemos tener en cuenta una serie de parámetros; puesto que al interrumpir bruscamente la corriente eléctrica se producen ciertos fenómenos que se tienen que conocer.

¿Qué debemos tener en cuenta cuando seleccionamos y utilizamos un interruptor?

a. Máximo voltaje que puede soportar cuando está abierto, ya que si se conecta un voltaje mayor al especificado, pueden producirse arcos de voltaje entre los contactos. Además una pequeñísima parte de contacto puede transportarse al otro, lo cual, al cabo de un determinado tiempo de conmutaciones puede destruir el dispositivo.

b. Máxima corriente que puede soportar cuando este cerrado, si la corriente que va a circular por el dispositivo es superior a la especificada, este se calienta y puede llegar a destemplarse los resortes que mantienen presionados los contactos, o los contactos pueden fundirse y dejar el circuito cerrado permanentemente.

c. Material de los contactos sea buen conductor y tenga alta resistencia a la temperatura y al desgaste mecánico, el cual se especifica en ciclos o numero de veces que se ha usado.

¿Cuáles son las fallas más comunes?

- Si el interruptor esta cerrado, o sea conduce en las dos posiciones, seguro que sus contactos se han fundido debido a un sobrecalentamiento producido por la circulación de una corriente elevada a través de él.

- Si por el contrario, el interruptor permanece abierto, pueden existir dos razones para ello.

* Las superficies de contactos se han desgastado impidiendo que haya un contacto físico cuando cambia de posición.

* Las superficies de contactos están recubiertas con una capa de óxido, la cual actúa como aislante.

Conectores. Según Talca (2014) son componentes electromecánicos, se usan para unir eléctricamente 2 o más circuitos dentro de un aparato o para conectar; junto con los cables apropiados, diferentes aparatos o dispositivos entre sí fácil y rápidamente. La mayoría de los conectores pueden ser empleados sin usar herramientas.

Fusibles. Según Talca (2014) son dispositivos empleados para proteger circuitos y sistemas electrónicos, de corriente excesivas y de cortos circuitos. Un fusible es una porción de alambre de muy buen conductor, cuya temperatura de fusión es mucho menor que la del cobre. Como el alambre es muy buen conductor, posee una resistencia muy baja y su longitud y diámetro son calculados para que se funda solo cuando circule por él una corriente mayor que aquella para la que fue diseñada.

¿Cómo elegir un fusible?

* Conociendo la corriente máxima por el circuito, debemos escoger un fusible cuya capacidad sea ligeramente superior; pero nunca mayor al 15%.

Dicho fusible puede calcularse.

Ejemplo:

$$I_{\max} = 400 \text{ mA}$$

$$Fus = I_{\max} \times 15\%$$

$$Fus = 400 \text{ mA} \times 1.15$$

$$Fus = 460 \text{ mA}$$

Diodos. De acuerdo a Vassallo (2000) el diodo es el dispositivo semiconductor más sencillo y se puede encontrar, prácticamente en cualquier circuito electrónico. Los diodos se fabrican en versiones de silicio (la más utilizada) y de germanio.

Está formado por la unión de dos cristales semiconductores, uno de tipo N, llamado cátodo, y otro de tipo P, llamado ánodo, separados por una barrera o unión. Esta barrera o unión es de 0.3 voltios en el diodo de germanio y de 0.6 voltios aproximadamente en el diodo de silicio.

Maneras de conectarse un diodo. - *Polarización directa:* Es cuando la corriente que circula por el diodo sigue la ruta de la flecha (la del diodo), o sea del ánodo al cátodo. En este caso la corriente atraviesa el diodo con mucha facilidad comportándose prácticamente como un cortocircuito.

- *Polarización inversa:* Es cuando la corriente en el diodo desea circular en sentido opuesto a la flecha (la flecha del diodo), o sea del cátodo al ánodo. En este caso la corriente no atraviesa el diodo, y se comporta prácticamente como un circuito abierto.

Tipos de diodos. Teniendo en cuenta a Vassallo (2000) se tienen los siguientes tipos de diodos.

- *Diodos rectificadores:* permiten la rectificación de la corriente alterna, transformándola en continua. Cuando están sometidos a polarización directa, conducen la corriente eléctrica a partir de una tensión de entre 0,2 y 0,8 V. Su encapsulado puede ser de plástico, de metal o cerámico, según su potencia. El cátodo viene señalado por un anillo blanco. Se identifica mediante un código alfanumérico.

- *Diodos de señal:* los diodos de señal de uso general se emplean en funciones de tratamiento de la señal, dentro de un circuito o bien para realizar operaciones de tipo digital formando parte de «puertas» lógicas y circuitos equivalentes, Son de baja potencia. El encapsulado es en forma de un cilindro miniatura, de plástico o vidrio, estando los dos terminales de conexión situados en los extremos. Sobre el cuerpo deberá estar marcado el hilo de conexión que corresponde al cátodo, mediante un anillo situado en las proximidades de éste.

- *Diodos de conmutación:* los diodos de conmutación o rápidos se caracterizan por ser capaces de trabajar con señales de tipo digital o <>lógico>> que presenten unos tiempos de subida y bajada de sus flancos muy breves. El factor o parámetro que caracteriza a estos diodos es el tiempo de recuperación inverso (TRR) que expresa el tiempo que tarda la unión P-N en desalojar la carga eléctrica que acumula, cuando se encuentra polarizada inversamente (efecto similar a la acumulación de carga de un condensador), y recibe súbitamente un cambio de tensión que la polariza en sentido directo. Pueden ser considerados rápidos aquellos diodos con un TRR inferior a 400 nanosegundos, en modelos de media potencia, para los de baja potencia este tipo es del orden de los 5 nanosegundos.

- *Diodos de alta frecuencia:* los diodos de alta frecuencia se emplean en aquellas partes de un circuito que deben de funcionar con frecuencias superiores a 1 megahertz (1 millón de ciclos por segundo). Se caracterizan por presentar una baja capacidad de difusión (Cd) entre las dos zonas semiconductoras que forman la unión P-N, cuando éstas están polarizadas en sentido directo.

- *Diodos zéner:* se denominan también reguladores de tensión porque su misión es estabilizar los valores de voltaje a que está sometido un circuito. Están diseñados para trabajar con tensiones inversas, es decir, en caso de polarización inversa. Se intercalan en los circuitos para que, si se produce este tipo de tensiones, la corriente se derive a través del diodo y no dañe al resto de los componentes del circuito. Se emplea en circuitos de control. Existe una amplia gama de tipos clasificados por una serie de tensiones zéner normalizadas y por la potencia que son capaces de disipar, desde 250 milivatios hasta decenas de vatios, con encapsulado plástico o metálico.

- *Diodos led:* emiten luz cuando son sometidos a polarización directa. Se emplean para señalización luminosa y para alumbrado. Precisan de una tensión mínima para emitir luz (de 1,5 a 2 V). Para conseguirla se intercala una resistencia en serie. El cátodo se identifica fácilmente observando el interior de la cápsula (lado plano) o la longitud de los terminales (terminal corto).

- *Diodos especiales:* dentro del grupo de diodos especiales están comprendidos diodos varicap, los fotodiódos, diodos túnel y diodos láser Los primeros se construyen buscando acentuar al máximo la propiedad que presente la unión P-N de comportarse de una forma análoga a un condensador, cuando se la polariza inversamente. La capacidad resultante es, además, variable con la tensión aplicada; lo cual permite disponer de una forma muy simple de condensadores variables, controlados por una diferencia de potencial. Su empleo está muy generalizado en etapas de sintonía de receptores de radio y TV.

Transistores. De acuerdo a Schilling y Charles (1991) es uno de los componentes electrónicos más versátiles. Está formado por la unión de tres cristales semiconductores. El funcionamiento del transistor está basado en la capacidad de gobernar la intensidad de corriente que circula entre el emisor y el colector mediante el paso de una pequeña corriente eléctrica por la base.

En los transistores comerciales, el emisor suele estar señalado por un punto o por un saliente en la cápsula. En otros casos, el emisor corresponde al terminal más largo. En la cápsula del transistor, aparecen marcadas las referencias que permiten identificar de qué tipo se trata. Existen dos clases de transistores: lo bipolares y los de efecto campo. Los transistores bipolares están formados por la unión de tres cristales semiconductores. Según la combinación de éstos, pueden ser de dos clases: NPN y PNP. De cada uno de los cristales sale un terminal que permite conectar físicamente el componente al circuito. Los terminales se denominan base (B), emisor (E) y colector (C).

Los transistores de efecto campo (FET) están formados por un sustrato de material semiconductor sobre el que se difunden dos islas de material semiconductor de diferente dopado. De las dos islas salen dos terminales que permiten conectar físicamente el componente al circuito. Los terminales se denominan Surtidor (S) y drenador (D). La conductividad entre estos dos terminales es gobernada por un tercer terminal llamado puerta (G), que está aislado eléctricamente del sustrato. Con el mismo principio de funcionamiento existe una amplia variedad de transistores por efecto campo. Los más abundantes, dado su amplio uso, son los MOS (Metal Óxido Semiconductor). Como los anteriores, también pueden ser de dos tipos: de canal N y de canal P.

Aplicaciones: amplificación de todo tipo (radio, televisión, instrumentación), generación de señal (osciladores, generadores de ondas, emisión de radiofrecuencia), conmutación, actuando de interruptores (control de relés, fuentes de alimentación conmutadas, control de lámparas, modulación por anchura de impulsos PWM) y detección de radiación luminosa (fototransistores).

Circuitos integrados. Teniendo en cuenta Schilling y Charles (1991): un circuito integrado es un circuito formado por elementos activos y pasivos tales como diodos, transistores, resistencias y condensadores, los cuales están interconectados y ubicados en una pastilla de silicio. Tienen dimensiones muy reducidas y sus elementos no se pueden separar, es decir, el sistema electrónico está formado por circuitos completos y cada uno de ellos contiene centenas de elementos, todos ellos situados en el cristal de silicio, algunos de los circuitos integrados más avanzados son los microprocesadores que controlan múltiples artefactos: desde computadoras hasta electrodomésticos, pasando por los teléfonos móviles. Otra familia importante de circuitos integrados la constituyen las memorias digitales.

El transistor actúa como un switch. Este puede encenderse electrónicamente o apagarse, o también puede amplificar corriente. Es utilizado por ejemplo en computadoras para almacenar

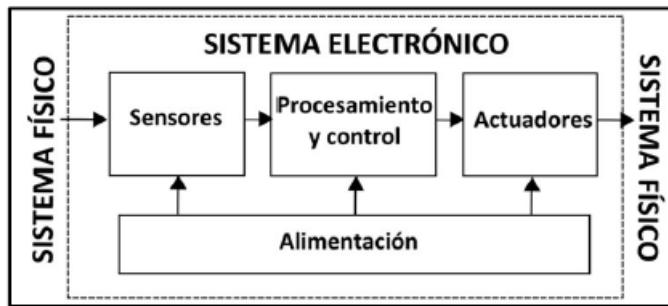
la información o en los amplificadores de un estéreo para hacer la señal del sonido más fuerte. Las resistencias limitan el flujo de electricidad y nos dan la posibilidad de controlar la cantidad de corriente que es permitida para fluir, las resistencias son utilizadas, entre otras cosas, para controlar el volumen en una televisión o en una radio. Los capacitores almacenan electricidad y la liberan en un rápido impulso, como en las cámaras fotográficas con una pequeña batería se puede provocar un fuerte flash para iluminar toda la habitación por un instante. Los diodos detienen la electricidad bajo alguna condición, y le permiten el paso tan solo cuando esta condición cambia. Esto es utilizado por ejemplo, en las foto celdas donde un haz de luz se corta y activa el diodo para detener el flujo a través de él. Estos componentes son como los bloques para armar en un circuito integrado, dependiendo de cómo son colocados los componentes se puede obtener desde una simple alarma hasta un complejo microprocesador de una computadora. El primer CI fue desarrollado en 1958 por el ingeniero Jack Kilby justo meses después de haber sido contratado por la firma Texas Instruments. Se trataba de un dispositivo que integraba seis transistores en una misma base semiconductor.

Tabla 7: Clasificación de circuitos integrados. Fuente: Schilling y Charles (1991)

Criterio	Tipo	Características
Según su tecnología de fabricación	Monolítico	Todos los componentes se construyen sobre el mismo cristal semiconductor o chip.
	Pelicular	Los componentes se van formando sobre la base de un sustrato aislante.
	Multilaminar	Los componentes se forman en capas diferentes y se unen a través de un sustrato común.
	Híbrido	Se utilizan todas las técnicas anteriores para la formación de los componentes.
Según el tipo de transistor empleado	Bipolar	Se forman a partir de transistores NPN o PNP y, según la forma de trabajo para la que estén diseñados, dan lugar a diferentes familias de circuitos: RTL, DTL, TTL, ECL, I^2L
	Unipolar	Se forman a partir de transistores MOSFET y, según sea el tipo de transistor utilizado, se clasifican en NMOS, PMOS y CMOS. Consumen menos corriente que los bipolares.
Según el grado de integración	SSI (Small Scale Integration)	Integración a pequeña escala (menos de 100 transistores por chip)
	MSI (Medium Scale Integration)	El número de transistores integrados en un solo chip varía entre 100 y 1000.
	LSI (Large Scale Integration)	Integración a gran escala (entre 1000 y 10000 transistores por chip)
	VLSI (Very Large Scale Integration)	En único chip se integran más de 10000 transistores.
Según su aplicación	Circuito integrado analógico o lineal	La señal de salida varía en el tiempo de acuerdo con la señal aplicada en la entrada. Dentro de un rango mínimo y máximo, puede tener infinitud de valores intermedios. Pueden constar desde simples transistores encapsulados juntos, sin unión entre ellos, hasta circuitos completos y funcionales, como amplificadores, osciladores o incluso receptores de radio completos.
	Circuito integrado digital	La señal de salida puede tener un valor mínimo (0 lógico) o un valor máximo (1 lógico), pero nunca un valor intermedio entre ambos.
	Circuito integrado específico	Realiza funciones que emplean tanto la tecnología analógica como la digital.

Sistema electrónico. Teniendo en cuenta a Artero (2013) un sistema electrónico es un conjunto de: sensores, circuitería de procesamiento y control, actuadores y fuente de alimentación. Los sensores obtienen información del mundo físico externo y la transforman en una señal eléctrica que puede ser manipulada por la circuitería interna de control. Existen sensores de todo tipo: de temperatura, de humedad, de movimiento, de sonido (micrófonos), etc. Los circuitos internos de un sistema electrónico procesan la señal eléctrica convenientemente. La manipulación de dicha señal dependerá tanto del diseño de los diferentes componentes hardware del sistema, como del conjunto lógico de instrucciones (es decir, del “programa”) que dicho hardware tenga pregrabado y que sea capaz de ejecutar de forma autónoma. Los actuadores transforman la señal eléctrica acabada de procesar por la circuitería interna en energía que actúa directamente sobre el mundo físico externo. Ejemplos de actuadores son: un motor (energía mecánica), una bombilla (energía lumínica), un altavoz (energía acústica), etc. La fuente de alimentación proporciona la energía necesaria para que se pueda realizar todo el proceso descrito de “obtención de información del medio, procesamiento y actuación sobre el medio”. Ejemplos de fuentes son las pilas, baterías, adaptadores AC/DC, etc.

Figura 20: Sistema electrónico. Fuente: Artero (2013)



Microcontrolador: Según Artero (2013, pp. 62-53) un microcontrolador es un circuito integrado o “chip” (es decir, un dispositivo electrónico que integra en un solo encapsulado un gran número de componentes) que tiene la característica de ser programable. Es decir, que es capaz de ejecutar de forma autónoma una serie de instrucciones previamente definidas por nosotros. En el diagrama anterior, representativo de un sistema electrónico, el microcontrolador sería el componente principal de la circuitería de procesamiento y control. Por definición, un microcontrolador (también llamado comúnmente “micro”) ha de incluir en su interior tres elementos básicos:

- *CPU (Unidad Central de Proceso):* es la parte encargada de ejecutar cada instrucción y de controlar que dicha ejecución se realice correctamente. Normalmente, estas instrucciones hacen uso de datos disponibles previamente (los “datos de entrada”), y generan como resultado otros datos diferentes (los “datos de salida”), que podrán ser utilizados (o no) por la siguiente instrucción.

- *Diferentes tipos de memorias:* son en general las encargadas de alojar tanto las instrucciones

como los diferentes datos que estas necesitan. De esta manera posibilitan que toda esta información (instrucciones y datos) esté siempre disponible para que la CPU pueda acceder y trabajar con ella en cualquier momento. Generalmente encontraremos dos tipos de memorias: las que su contenido se almacena de forma permanente incluso tras cortes de alimentación eléctrica (llamadas “persistentes”), y las que su contenido se pierde al dejar de recibir alimentación (llamadas “volátiles”). Según las características de la información a guardar, esta se grabará en un tipo u otro de memoria de forma automática, habitualmente.

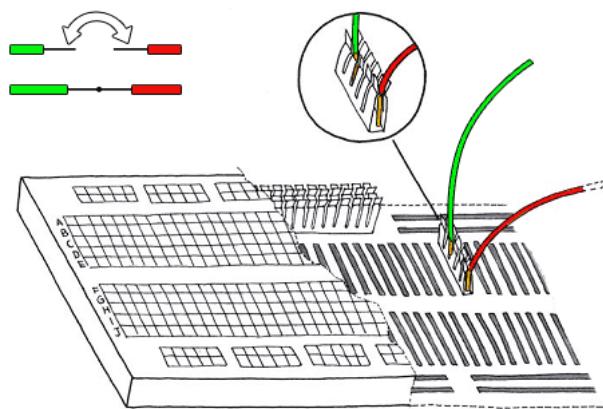
- *Diferentes patillas de E/S (entrada/salida)*: son las encargadas de comunicar el microcontrolador con el exterior. En las patillas de entrada del microcontrolador podremos conectar sensores para que este pueda recibir datos provenientes de su entorno, y en sus patillas de salida podremos conectar actuadores para que el microcontrolador pueda enviarles órdenes y así interactuar con el medio físico. De todas formas, muchas patillas de la mayoría de microcontroladores no son exclusivamente de entrada o de salida, sino que pueden ser utilizados indistintamente para ambos propósitos (de ahí el nombre de E/S).

Señales de un sistema electrónico. - *Señal analógica*: Puede tomar cualquier valor de amplitud; la amplitud varia continuamente en el tiempo; la señal obtenida por un transductor es analógica; mas sensibles al ruido (perturbaciones); mas difíciles de implementar en circuitos integrados.

- *Señal digital*: Solo toma un numero finito de amplitudes; su amplitud es en lógica binaria dos; la amplitud cambia en instantes espaciados uniformemente; menos sensibles al ruido (perturbaciones); mas fáciles de implementar en circuitos integrados.

Protoboard Teniendo en cuenta a Talca (2014) el tablero para prototipos o tablero de pruebas para conexiones es un dispositivo muy ingenioso que nos permite armar/desarmar rápida y fácilmente cualquier circuito electrónico, sin necesidad de soldaduras, y muchas veces sin herramientas. En electrónica, esto es muy útil durante el proceso de aprendizaje para hacer experimentos y proyectos en forma provisional. Estos tableros están formados por una base de plástico que tiene una serie de perforaciones con una disposición especial. Debajo de estas perforaciones se encuentran unas láminas metálicas que forman contactos en forma vertical u horizontal, en donde se unen los diferentes componentes electrónicos de los circuitos. Éstas láminas son fabricadas con un metal flexible de berilio-cobre recubierto con plata-níquel, y en algunos casos de oro. El recubrimiento impide que los contactos se oxiden y la flexibilidad del metal permite utilizar cables y terminales de diferente diámetro, sin deformarse.

Figura 21: Protoboard. Fuente: Plastica (2018)



Fundamentos de la informática. Según del Prado y Lamas (2013, pp. 4) «el término informática ha ido evolucionando a lo largo del tiempo, pero en la actualidad se considera la ciencia que estudia el tratamiento automático de la información».

Estructuras de un sistema de computación. De acuerdo a del Prado y Lamas (2013, pp. 4) «un sistema de computación consiste de uno o más procesadores, memoria principal, relojes, terminales, discos, interfaces de red y otros dispositivos de entrada/salida. Sin embargo, hardware sin software es simplemente inútil. El sistema de operación es una parte importante de un sistema de computación».

Para realizar este tratamiento de la información nuestros sistemas informáticos deben llevar a cabo las siguientes tareas básicas de la información: entrada de datos (captación), procesamiento/tratamiento y salida de datos (transmisión de resultados), éstas tareas es posible gracias a las partes físicas de nuestros sistemas informáticos (hardware), la entrada y salida de datos se realizará mediante los periféricos de nuestro sistema y el tratamiento de la información se realizará mediante la unidad central de procesamiento (CPU).

Componentes de un sistema:

CPU (procesador) Unidad central de procesamiento (procesador): Permite ejecutar un conjunto de instrucciones. Su velocidad es varios órdenes mayor con respecto al acceso a la memoria.

Memoria: Permite mantener la información disponible. Existe una jerarquía de memoria: registros, caches, memoria física de tipo RAM (Random Access Memory), dispositivos magnéticos, ópticos.

Dispositivos de Entrada/Salida (IO): Permiten interactuar con el sistema. Algunos dispositivos más comunes: impresoras, teclados, ratón, video, disco, red, etc.

Hardware. Según del Prado y Lamas (2013) comprende todos los dispositivos o elementos físicos (que se pueden tocar) con los cuales es construida una computadora. Incluye también

los elementos mecánicos, electrónicos y eléctricos. Los teclados, monitores, impresoras, microprocesadores, unidades de disco, ratón, escáner y demás periféricos, son hardware.

Placa Base. Teniendo en cuenta a Mas (2011a) la placa base es el circuito impreso al que se conectan todos los componentes de un computador como:

- *Zócalo de CPU (Socket)*: conecta el procesador a la placa, lo que permite seleccionar el procesador que mejor se adapta a nuestras necesidades y sustituirlo en caso de avería o ampliación. Aunque cada zócalo sólo admite una determinada familia de CPU's. Ejemplos: los procesadores AMD (Socket AM2 [Athlon 64, FX, X2, Sempron, ...], Socket AM3 [Phenom II, X2/X3/X4]) y los procesadores Intel (Socket 775 [Pentium 4, Celeron, Core2, ...], Socket 1156 [Core i3, Core i5, Core i7, ...])

- *Chipset*: es el conjunto de circuitos que componen la placa base, es el foco de atención para decidirse por una placa, también el fabricante de la placa, naturalmente, aunque, en general, éste se limita a seleccionar el chipset y montar sus circuitos. El chipset determina (entre otras): el tipo de CPU (o familia) que se pueden montar en la placa, muchas características importantes (soporte específico para determinados componentes del ordenador, tecnología, conectividad y compatibilidad con otros dispositivos, memoria máxima, posibilidades de ampliación, etc), la velocidad con que se comunican entre sí los componentes del ordenador. En resumen, determina la calidad (y el precio) de la placa.

Un chipset se compone de dos partes:

* *Puente Norte (Northbridge)* es el circuito que comunica la CPU con la memoria RAM, con la tarjeta de vídeo y con el puente sur. Es el circuito más importante de un ordenador (después de la CPU, claro está). Algunas CPU's de última generación lo incorporan en sus propios circuitos.

* *Puente Sur (Southbridge)* es el circuito que coordina tanto los dispositivos de entrada-salida como cualquier otro dispositivo conectado a la placa través de las ranuras de expansión (PCI, USB, IDE, SATA, sonido, ...).

- *Ranuras de memoria*: permiten la conexión de memoria RAM a la placa, establece la memoria máxima del sistema, el número de ranuras en una placa suele oscilar entre 2 y 6, las hay de varios tipos, se diferencian por el número de contactos y la disposición de éstos, sólo en un lado o en ambos lados: SIMM (sólo en un lado, 32 bits, en desuso) [Módulos de 30 ó de 72 contactos], DIMM (en ambos lados, 64 bits) [SDR (168 contactos), DDR (184 contactos), DDR2 (240 contactos), DDR3 (240 contactos), DDR4 (288 contactos)], SO DIMM (32 y 64 bits), utilizado en los portátiles (144 contactos).

- *Conectores de entrada/salida*: los hay internos (no accesibles desde el exterior) y externos, permiten la conexión de distintos dispositivos a la placa (Discos duros, Periféricos, Dispositivos de red...), los tipos más comunes; VGA, DVI, HDMI, USB, IDE o Parallel ATA (PATA), Serial ATA (SATA), eSATA (conector SATA externo), RJ45 (ethernet), Firewire, Puerto serie, paralelo, PS/2, etc.

- *Ranuras de expansión*: permiten ampliar la capacidad del ordenador conectando otras

tarjetas a la placa (Red, Video, Audio, etc.), los tipos más comunes; ISA (escaso ancho de banda, en desuso, reemplazado por PCI), AGP (sólo para tarjetas de video, insuficiente ancho de banda para la tecnología actual, en desuso, reemplazado por PCI Express), PCI (resultó muy eficaz en tecnología “plug and play”, 32 bits), PCI Express (evolución de PCI: 64 bits, transferencia de datos en serie mediante el uso de múltiples canales: PCI Express x1, PCI Express x2, x4, x8, x16).

- *Video/Sonido/Red integrados*: algunas placas llevan estas funcionalidades integradas, en realidad, se trataría de una característica del chipset de la placa (aunque no siempre es un valor añadido).

- *CMOS*: es la memoria que almacena la configuración del equipo cuando éste está apagado

- *BIOS*: Sistema básico de entrada y salida para arrancar el ordenador

- *Buses de comunicación*: transportan la información entre componentes de la placa.

Unidad Central de Proceso (CPU). Según Mas (2011a) Interpreta y procesa las instrucciones de los programas, características principales; rango de enteros 32 / 64 bits, frecuencia de reloj (velocidad) [2.4 GHz, 3 GHz, ...], fabricante y modelo; Intel vs. AMD (Pentium 4, Core i5, Phenom, ...), Núcleos; Dual-core, Quad-core, Hexa-core, etc.

Teniendo en cuenta a del Prado y Lamas (2013):

- *Unidad de Control (UC)*: interpreta las instrucciones y gobierna la ejecución de las mismas, gestionando el modo en que se controlan los diversos elementos del computador. La UC genera todas las señales que forman el bus de control en función de la información que dispone de: la propia instrucción (operación a realizar, modo de direccionamiento, etc.), el registro de estado, interfaz con memoria, interfaz con el sistema de E/S, el comportamiento de un computador es síncrono. La ejecución de una instrucción está gobernada por un contador de períodos.

- *Unidad Aritmético Lógica (UAL)*: es la parte de la UCP encargada de realizar operaciones aritméticas y lógicas sobre la información. Las operaciones aritméticas pueden ser suma, resta, multiplicación, división, potenciación, etc. Las lógicas son normalmente de comparación, para las que se emplean los operadores del álgebra de Boole.

- *Operacional o circuito operacional*: realiza las operaciones con los datos de los registros de entrada.

- *Registros de entrada*: contienen los operandos de la operación.

- *Acumulador*: almacena los resultados de las operaciones.

- *Registro de estado*: registra las condiciones de la operación anterior.

- *Buses*: es el elemento responsable de establecer una correcta interacción entre los diferentes componentes del ordenador. Es, por lo tanto, el dispositivo principal de comunicación. En un sentido físico, se define como un conjunto de líneas de hardware (metálicas o físicas) utilizadas para la transmisión de datos entre los componentes de un sistema informático. En cambio, en sentido figurado es una ruta compartida que conecta diferentes partes del sistema. La evolución de los buses a lo largo de la historia ha sido determinante para la evolución

de los sistemas operativos. Al fabricarse buses mucho más rápidos y con más líneas, los sistemas operativos han ido mejorando y aportando nuevas funcionalidades que antiguamente no podían ser implementadas por falta de velocidad en los buses. Hoy por hoy, el bus sigue determinando en gran medida la velocidad de proceso de un equipo, ya que sigue siendo uno de los componentes hardware que mayores limitaciones tiene.

Memoria: Funciones, tipos y características. Según Mas (2011a) las unidades de capacidad son:

1 Bit = 0 ó 1 (unidad mínima de información)

1 Byte= 8 Bits

1 Kilobyte(KB) = 1024 (o 1000) Bytes

1 Megabyte(MB) = 1024 (o 1000) Kilobytes

1 Gigabyte(GB) = 1024 (o 1000) Megabytes

1 Terabyte(TB) = 1024 (o 1000) Gigabytes

Petabyte, Exabyte, Zettabyte, Yottabyte, Brontobyte, ...

De acuerdo a del Prado y Lamas (2013) el ordenador almacena dentro de su memoria interna todos los programas y datos con los que se va a trabajar y que van a ser procesados.

Los dos tipos de memoria esenciales con los que puede trabajar el ordenador son:

- *Memorias de almacenamiento externo:* de acuerdo a del Prado y Lamas (2013) se les da esta denominación a los soportes de almacenamiento, ya que son capaces de almacenar información. Son memorias externas: discos duros, disquetes, cintas DAT, pendrives, tarjetas SD, tarjetas microSD, etc. Estas memorias son más lentas que la propia memoria principal, ya que constan de componentes electrónicos y mecánicos. Son no volátiles, de tal forma que la información permanece en ellas incluso después de quitar el suministro de energía eléctrica al ordenador.

- *Memoria de almacenamiento interna:* de acuerdo a del Prado y Lamas (2013) dentro del ordenador existen varios tipos de memorias que no son consideradas externas.

Teniendo en cuenta a de acuerdo a del Prado y Lamas (2013) los tipos de memorias internas:

* RAM (Random Access Memory). En ella es posible almacenar y modificar información y es lo que se conoce como memoria principal, memoria central o memoria de acceso directo.

Tipos de Memoria RAM:

DRAM (Dynamic RAM). Es un tipo de memoria RAM electrónica construida mediante condensadores. Cuando un condensador está cargado se dice que almacena un BIT a uno. Si está descargado, el valor del BIT es cero. Para mantener las celdillas cargadas, este tipo de memoria necesita refrescarse cada cierto tiempo: el refresco de una memoria RAM consiste en recargar nuevamente con energía los condensadores que tienen almacenado un uno para evitar que la información se pierda (de ahí lo de Dynamic). La memoria DRAM es más lenta que la memoria SRAM, pero mucho más barata de fabricar.

SRAM (Static RAM). Es un tipo de memoria RAM alternativa a la DRAM que no necesita refrescarse. SRAM y DRAM son memorias volátiles, lo que significa que cuando se corta el

suministro de corriente, los datos almacenados se pierden. Debido al alto coste de fabricación de la SRAM y a su alta velocidad, suele utilizarse como memoria caché.

SDRAM (Synchronous Dynamic RAM). Es una memoria que incorpora la capacidad de la DRAM y la velocidad de la SRAM; es decir, necesita refresco de sus celdas, pero en un intervalo superior de tiempo. Esta memoria es la que incorporan en la actualidad la mayoría de los ordenadores personales.

DDRAM (Double Data Rate) o memoria de doble recarga o memoria de doble tasa de transferencia. Compuesta por memorias SDRAM, tiene la característica de que se refresca dos veces por impulso de reloj. Es una memoria de funcionamiento muy complejo, pero tiene la ventaja de ser prácticamente el doble de rápida que cualquiera de las anteriores. En la actualidad, una de las características fundamentales de las memorias RAM es la velocidad con que la información se puede almacenar en ellas. Esta velocidad es mayor cuanto menos se tarde en acceder a la posición de memoria requerida en cada instante. La velocidad se mide en nanosegundos (60, 70, 80, 100,...). Cuanto menor sea el tiempo de acceso, más rápido será el acceso que se pueda realizar a cualquier posición de memoria para poder grabar o leer su información.

* ROM (Read Only Memory). Es una memoria de solo lectura, cuya información no puede ser modificada y que sirve básicamente para poder inicializar el sistema informático. La memoria ROM contiene programas especiales que sirven para cargar e iniciar el arranque del ordenador. En ella se encuentra almacenada toda la información referente a los componentes hardware de los que consta nuestro equipo. Posteriormente, será labor del sistema operativo realizar el resto de operaciones para poder empezar a utilizar el ordenador. El software que integra la ROM forma el BIOS del ordenador (Basic Input Output System) o sistema básico de entrada/salida. El BIOS se encuentra físicamente en varias partes del ordenador. El componente principal está en la placa base. Inicialmente, los BIOS se programaban sobre memorias de tipo ROM, lo que implicaba que cualquier modificación en el sistema no podía realizarse a menos que lo hiciese el fabricante. Había que sustituir el componente electrónico para modificar la configuración del BIOS. Por eso, posteriormente, el BIOS se montó en memorias de tipo PROM (Programmable Read Only Memory), que son programables una sola vez y después de haber sido montadas en la placa. El BIOS es un código que localiza y carga el sistema operativo en la RAM; es un software elemental instalado en una pequeña ROM de la placa base que permite que esta comience a funcionar. Proporciona las órdenes básicas para poner en funcionamiento el hardware indispensable para empezar a trabajar. Como mínimo, maneja el teclado y proporciona salida básica (emitiendo pitidos normalizados por el altavoz del ordenador si se producen fallos) durante el arranque.

En la actualidad, se utilizan las memorias de tipo EPROM (Erasable Programmable Read Only Memory), que permiten cambiar la configuración asignada. Este proceso es complejo, pero no implica realizar operaciones físicas sobre los componentes que están montados. Todas estas memorias son no volátiles, y la información que contienen no desaparece nunca debido a

que están programadas de fábrica. No necesitan ningún suministro de energía para mantener su configuración. La CMOS (Complementary Metal Oxide Semiconductor) es un tipo de memoria interna del ordenador que se caracteriza por consumir muy poca energía eléctrica, lo que la hace idónea para almacenar datos del BIOS. El BIOS es un tipo de memoria que en los ordenadores se utiliza para guardar los datos básicos de hardware y de configuración. Por ejemplo, en él se guarda la información sobre los discos duros (cuántos y de qué características) y otras informaciones como la fecha y la hora. Para que toda la información que mantiene no se borre, es necesario que la CMOS siempre tenga corriente eléctrica. Cuando el ordenador está apagado, obtiene energía de una pequeña pila o batería ubicada en la placa base. La configuración del BIOS se puede modificar si instalamos un nuevo disco duro, si queremos cambiar la fecha, la hora del sistema, etc. Para acceder al BIOS y poder modificar sus valores, hay que pulsar las teclas F2 o Supr durante el proceso de inicio del equipo, dependiendo del BIOS de cada equipo. Así se accede al SETUP del equipo, en el que se configuran las opciones de inicio básicas del ordenador.

En cuanto a la estructura de la memoria, desde un principio, e independientemente del sistema operativo. La memoria se ha estructurado en varios niveles; Memoria convencional (0 a 640 KB), Memoria superior (641 KB hasta 1024 KB), Memoria extendida (1024 KB hasta el límite de la placa base del equipo). Los fabricantes de ordenadores han dividido desde un principio la estructura en esas tres partes fundamentales, que son gestionadas de forma diferente por cada sistema operativo.

Dispositivos de Entrada/Salida. Teniendo en cuenta a del Prado y Lamas (2013) la unidad de entrada/salida sirve para comunicar el procesador y el resto de componentes internos del ordenador con los periféricos de entrada/salida y las memorias de almacenamiento externo o auxiliar. Los periféricos son dispositivos hardware con los cuales el usuario puede interactuar con el ordenador (teclado, ratón, monitor), almacenar o leer datos y/o programas (dispositivos de almacenamiento o memorias auxiliares), imprimir resultados (impresoras), etcétera. Los dispositivos de entrada son aquellos que nos permiten introducir datos y programas en el ordenador desde el exterior hacia su memoria central para que puedan ser utilizados, modificada o simplemente guardados. Entre estos tenemos scanner, teclado, mouse, micrófono. Los dispositivos de salida nos permiten extraer información desde el ordenador hacia el exterior, como ejemplo tenemos el monitor, la impresora, plotter, etc. Los dispositivos de almacenamiento son aquellos que nos permiten guardar la información de una computadora que se genera dentro de una computadora para luego volver a modificarla o verla, como ejemplo de estos dispositivos tenemos el Disco Duro, Unidad de DVD, Unidad de CD, Memorias USB, etc.

Muchos de los periféricos de entrada/salida necesitan un tipo de software especial para ser configurados; en otras palabras, para utilizar una impresora, por ejemplo, primero hay que instalarla. Esto significa que es necesario introducir dentro de la configuración del ordenador

y acorde con nuestro software básico unos programas específicos que permitan al sistema operativo reconocer el periférico y utilizarlo de forma correcta. Estos programas se denominan drivers o controladores.

Software. Según del Prado y Lamas (2013) es el soporte lógico e inmaterial que permite que la computadora pueda desempeñar tareas inteligentes, dirigiendo a los componentes físicos o hardware con instrucciones y datos a través de diferentes tipos de programas. Los programas nos servirán para procesar datos (información). El conjunto de varios programas se denomina aplicación informática. El sistema operativo es el componente software de un sistema informático capaz de hacer que los programas (software) procesen información (datos) sobre los componentes electrónicos de un ordenador o sistema informático (hardware). El sistema informático es el conjunto de elementos físicos o hardware que son necesarios para la explotación de las aplicaciones informáticas o software. Entre software y hardware existe otro concepto importante dentro de un sistema informático: el firmware, es el software introducido en componentes electrónicos o hardware.

Clasificación del software. Teniendo en cuenta a del Prado y Lamas (2013) el software: son los programas de aplicación y los sistemas operativos, que según las funciones que realizan pueden ser clasificados en:

- *Software de Sistema o de Base:* conjunto de programas que sirven para interactuar con el sistema, confiriendo control sobre el hardware, además de dar soporte a otros programas.

De acuerdo a del Prado y Lamas (2013) se divide en:

* *Sistema operativo:* es un conjunto de programas que administran los recursos de la computadora y controlan su funcionamiento.

Un Sistema Operativo realiza las siguientes funciones básicas según Mas (2011b):

1. Interfaz de usuario: Gráfico o línea de comando.
2. Núcleo del sistema (kernel): Manejo de interrupciones, de trabajo de la CPU y de comunicación.
3. Administrador de memoria: Asigna posiciones en la RAM a los programas.
4. Sistema de entrada/salida (E/S): Muestra al usuario la E/S de datos independiente del dispositivo.
5. Administrador de archivos: Supervisa la creación, actualización y eliminación de archivos.
6. Administrador de procesos: Asigna tiempos y organiza los procesos evitando conflictos entre ellos.

* *Controladores de Dispositivos:* son programas que permiten a otros programas de mayor nivel como un sistema operativo interactuar con un dispositivo de hardware.

* *Programas Utilitarios:* realizan diversas funciones para resolver problemas específicos, además de realizar tareas en general y de mantenimiento. Algunos se incluyen en el sistema operativo.

- *Software de Aplicación*: son los programas diseñados para o por los usuarios para facilitar la realización de tareas específicas en la computadora, de acuerdo a Mas (2011b) se clasifican las aplicaciones en: ofimáticas, científicas, gráficas, comunicaciones e Internet, seguridad, multimedia, juegos y educativos, utilidades de sistema o aplicaciones a medida.

Con respecto a las Licencias de aplicaciones de acuerdo a Mas (2011b) existen del tipo:

Libre: Dependiendo de la licencia marcada por el creador, el usuario puede tener la libertad para ejecutar, realizar copias, distribuir, estudiar, cambiar, mejorar o incluso cobrar el software. No necesariamente es gratuito. Ejemplo: General Public License (GPL), Berkeley Software Distribution (BSD), entre otros.

Propietario: El creador establece a través de diferentes tipos de licencia las limitaciones que el usuario tiene en el uso, modificación y distribución del software. No necesariamente de pago. Ejemplo: Shareware, Freeware, entre otros.

- *Software de Programación*: según del Prado y Lamas (2013) es el conjunto de herramientas que permiten al desarrollador informático escribir programas usando diferentes alternativas y lenguajes de programación. Este tipo de software incluye principalmente compiladores, intérpretes, ensambladores, enlazadores, depuradores, editores de texto y un entorno de desarrollo integrado que contiene las herramientas anteriores, y normalmente cuenta una avanzada interfaz gráfica de usuario (GUI).

Teniendo en cuenta a Mas (2011b) se pueden definir los siguientes términos en referencia a los software de programación:

* *Algoritmo*: Secuencia finita de reglas que, ejecutadas de forma secuencial, permite la realización de una tarea concreta, siendo válido para cualquier posibilidad de estado de los valores de entrada.

* *Programa*: La escritura de un algoritmo (o conjuntos de algoritmos) en un lenguaje inteligible para la máquina se denomina programa (instrucción como unidad básica).

* *Lenguaje de programación*: Los programas se escriben en un lenguaje concreto, posee un alfabeto, un vocabulario y un conjunto de reglas que definen la sintaxis y la semántica de las instrucciones. Se distingue entre lenguajes de bajo nivel y de alto nivel.

a. *Lenguajes de bajo nivel*: Actualmente, uso en situaciones muy concretas.

- *Lenguaje o código máquina*: Codificado en binario (presentación en hexadecimal), instrucciones directamente comprensibles por el ordenador, con lo que no necesita traducción pero escribir el programa no es nada rápido ni sencillo, velocidad y nivel de optimización altos, cada procesador tiene su propio código máquina.

- *Lenguaje ensamblador*: surge en 1945, lenguaje simbólico en el que cada instrucción corresponde directamente con una o varias instrucciones en código máquina, mnemotécnicos en lugar de las cadenas de bits, facilita algo la escritura de los programas, necesita traducción.

b. *Lenguajes de alto nivel*: los problemas de los lenguajes de bajo nivel son; dificultad de escritura, gran número de errores y conocimiento de la arquitectura concreta del procesador, las pretensiones del lenguaje de alto nivel; hacer que el programa sea independiente de la

máquina, acercar el lenguaje al lenguaje natural, para minimizar los errores, incluir un conjunto de procedimientos y rutinas que se utilicen comúnmente (librerías de funciones) para evitar reprogramarlas. La traducción de los lenguajes de programación: transformación del código escrito (fuente) a código máquina: tenemos a los;

- Ensambladores: Traducen el código ensamblador a código máquina.
- Compiladores: Traducen código en lenguajes de alto nivel a código máquina. Producen ejecutables estáticos (si se cambia algo hay que volver a compilar).
- Intérpretes: Realizan la traducción y la ejecución instrucción por instrucción. La ejecución es más lenta pero se puede detener y modificar la siguiente instrucción. Son muy robustos y fácilmente transportables (lenguajes independientes de la máquina)

Lenguajes de Programación.

Lenguaje C. Según Marzal y Gracia (2010) la primera versión «estable» del lenguaje data de 1978 y se conoce como «K&R C», es decir, «C de Kernighan y Ritchie». Esta versión fue descrita por sus autores en la primera edición del libro «The C Programming Language». La adopción de Unix como sistema operativo de referencia en las universidades en los años 80 popularizó enormemente el lenguaje de programa. C ha tenido un gran impacto en el diseño de otros muchos lenguajes. Ha sido, por ejemplo, la base para definir la sintaxis y ciertos aspectos de la semántica de lenguajes tan populares como Java y C++. El lenguaje de programación C es uno de los más utilizados (si no el que más) en la programación de sistemas software. Es similar a Python en muchos aspectos fundamentales: presenta las mismas estructuras de control (selección condicional, iteración), permite trabajar con algunos tipos de datos similares (enteros, flotantes, secuencias), hace posible definir y usar funciones, etc. No obstante, en muchas otras cuestiones es un lenguaje muy diferente. C presenta ciertas características que permiten ejercer un elevado control sobre la eficiencia de los programas, tanto en la velocidad de ejecución como en el consumo de memoria, pero a un precio: tenemos que proporcionar información explícita sobre gran cantidad de detalles, por lo que generalmente resultan programas más largos y complicados que sus equivalentes en Python, aumentando así la probabilidad de que cometamos errores.

C es un lenguaje compilado: antes de ejecutar un programa escrito por nosotros, suministramos su código fuente (en un fichero con extensión «.c») a un compilador de C. El compilador lee y analiza todo el programa. Si el programa está correctamente escrito según la definición del lenguaje, el compilador genera un nuevo fichero con su traducción a código de máquina, y si no, muestra los errores que ha detectado. Para ejecutar el programa utilizamos el nombre del fichero generado.

Python. De acuerdo a Ojeda (2015) fue creado por Guido van Rossum a principios de los años 90 en Holanda. El nombre del lenguaje proviene de la afición de su creador por el grupo de humoristas británicos los Monty Python. Python es un interpretador de instrucciones que permite usar el lenguaje en forma interactiva. Los lenguajes interpretados, a diferencia de los lenguajes compilados, permiten experimentar interactivamente en una ventana y también mediante programas que pueden desarrollarse y probarse a medida que son construidos. Esta interacción facilita el aprendizaje del lenguaje y mejora la productividad. Los programas compilados en cambio, deben estar completos para que sean probados y no admiten experimentar separadamente con las instrucciones. La ventaja de los programas compilados es que el tiempo de ejecución es menor. Python es un lenguaje de propósito general. Su diseño no obliga a los usuarios a adoptar un estilo particular. Esta característica del lenguaje motiva la creatividad y permite la elección entre varios paradigmas o metodologías de programación. Con todos los recursos del lenguaje y el soporte de las librerías disponibles, el programador puede usar libremente su imaginación para crear nuevas soluciones. Partiendo de un conocimiento inicial básico, puede avanzar en el aprendizaje del lenguaje a su propio ritmo. Python es un producto público y de distribución libre que puede descargarse de internet.

Características del lenguaje computacional Python

- a) Python es un lenguaje interpretado. Se considera sucesor del lenguaje ABC y usa conceptos de otros lenguajes como Modula-3, Lisp, entre otros.
- b) Python no obliga a los programadores a adoptar un estilo particular de programación.
- c) Se puede instalar en varias plataformas: Windows, Linux, etc. Con menores cambios puede trasladarse entre ellas.
- d) Es software libre y de código abierto con licencia GPL (General Public License). Se puede instalar, modificar y distribuir proporcionando el código fuente. Una licencia GPL no ofrece garantía, pero la gran comunidad de usuarios que disponen del código abierto, rápidamente detectan errores.
- e) El código escrito en Python es legible, sin marcas para definir bloques como en otros lenguajes. No requiere símbolos de fin de línea. Escribir en este lenguaje es casi como escribir en seudo código en inglés
- f) Usa el encolumnamiento de instrucciones para definir bloques y establecer el alcance de las estructuras de control. El código resultante es compacto pero legible.
- g) Python es un lenguaje de tipado dinámico: conecta un método y un nombre de variable durante la ejecución del programa.
- h) Es un lenguaje seguro. Realiza chequeo dinámico de tipos de datos y de índices.
- i) Es un lenguaje extendible. Continuamente la gran comunidad de usuarios está creando nuevas librerías en diferentes áreas de aplicación.
- j) Python es un lenguaje de propósito general. Algunas aplicaciones son: aprendizaje de la programación, desarrollo de prototipos, computación científica y matemática, estadística y optimización, desarrollo de interfaz visual, desarrollo de aplicaciones web, interfaz para manejo

de bases de datos, educación y juegos, desarrollo de software.

- k) Algunos usuarios actuales de Python: YouTube, Google, NASA, universidades, etc.
- l) Aprendizaje fácil para diferentes niveles de usuarios. Recomendable como primer lenguaje de programación.
- m) Python puede tener abiertas varias ventanas interactivas y varias ventanas de edición trabajando en pruebas o proyectos diferentes.
- n) En Python la ejecución se inicia desde la ventana de edición o cargando el programa en la ventana interactiva. Siendo un interpretador de lenguaje, el traductor debe estar presente. Sin embargo ya existen utilitarios para compilar programas Python. Ejemplo. PyPy.
- o) En Python, la mayoría de las funciones están en librerías que deben ser cargadas antes de acceder a las funciones. Por este motivo, el tiempo de carga del traductor Python es menor.
- p) Python es un lenguaje de propósito general y tiene estructuras de datos muy flexibles. Se pueden crear listas con componentes de diferentes tipos. La filosofía de Python es la legibilidad y la transparencia. Estos principios están codificados y se los puede revisar escribiendo la instrucción import this en la ventana principal de Python.

Php. Teniendo en cuenta Achour *et al.* (2019) PHP ha recorrido un largo camino desde su nacimiento a mediados de los 90. Desde sus humildes comienzos hasta ser uno de los más importantes lenguajes de programación web. PHP (acrónimo de "PHP: Hypertext Preprocessor") es un lenguaje de "código abierto" interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor. El código PHP se incluye entre etiquetas especiales de comienzo y final «<?php ?>» que nos permitirán entrar y salir del modo PHP. Lo que distingue a PHP de la tecnología Javascript, la cual se ejecuta en la máquina cliente, es que el código PHP es ejecutado en el servidor. Si tuviésemos un script similar al de nuestro ejemplo en nuestro servidor, el cliente solamente recibiría el resultado de su ejecución en el servidor, sin ninguna posibilidad de determinar qué código ha producido el resultado recibido. El servidor web puede ser incluso configurado para que procese todos los archivos HTML con PHP, además con PHP tiene la libertad de elegir el sistema operativo y el servidor de su gusto. También tiene la posibilidad de usar programación procedural o programación orientada a objetos. Aunque no todas las características estándar de la programación orientada a objetos están implementadas en la versión actual de PHP, muchas bibliotecas y aplicaciones grandes (incluyendo la biblioteca PEAR) están escritas íntegramente usando programación orientada a objetos. Con PHP no se encuentra limitado a resultados en HTML. Entre las habilidades de PHP se incluyen: creación de imágenes, archivos PDF y películas Flash (usando libswf y Ming) sobre la marcha. También puede presentar otros resultados, como XHTM y archivos XML. PHP puede autogenerar éstos archivos y almacenarlos en el sistema de archivos en vez de presentarlos en la pantalla. Quizás la característica más potente y destacable de PHP es su soporte para una gran cantidad de bases de datos (MySQL, Postgres, Oracle, Sybase, Sqlite, entre otros).

JavaScript. Según Docs (2019) JavaScript (JS) es un lenguaje ligero e interpretado, orientado a objetos con funciones de primera clase, más conocido como el lenguaje de script para páginas web, pero también usado en muchos entornos sin navegador, tales como node.js, Apache CouchDB y Adobe Acrobat. Es un lenguaje script multi-paradigma, basado en prototipos, dinámico, soporta estilos de programación funcional, orientada a objetos e imperativa. El estándar de JavaScript es ECMAScript. Desde el 2012, todos los navegadores modernos soportan completamente ECMAScript 5.1. Los navegadores más antiguos soportan por lo menos ECMAScript 3. El 17 de Julio de 2015, ECMA International publicó la sexta versión de ECMAScript, la cual es oficialmente llamada ECMAScript 2015, y fue inicialmente nombrada como ECMAScript 6 o ES6. Desde entonces, los estándares ECMAScript están en ciclos de lanzamiento anuales. JavaScript no debe ser confundido con el lenguaje de programación Java. Ambos "Java" y "Javascript" son marcas registradas de Oracle en Estados Unidos y otros países. Sin embargo, los dos lenguajes de programación tienen muchas diferencias en las sintaxis, semántica y usos. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico, además cuenta con una basta cantidad de librerías como Jquery o framework que facilitan la creación de funcionalidades como Angular, jQuery Mobile, entre otros.

Entorno de desarrollo integrado (IDE). Según Fuentes *et al.* (2001) un Entorno de Desarrollo Integrado (IDE) es una aplicación visual que sirve para la construcción de aplicaciones a partir de componentes. Por lo general todas ellas cuentan con los siguientes elementos: una o más ‘paletas’ para mostrar como iconos los componentes disponibles; un ‘lienzo’ o ‘contenedor’ en el cual se colocan los componentes y se interconectan entre sí; editores específicos para configurar y especializar los componentes; visores (browsers) para localizar componentes de acuerdo a ciertos criterios de búsqueda; directorios de componentes; acceso a editores, intérpretes, compiladores y depuradores para desarrollar nuevos componentes; y finalmente, acceso a algunas herramientas de control y gestión de proyectos y CSCW, esenciales para grandes proyectos software.

Ejemplos de entornos de desarrollo integrado (IDE); Netbeans, Eclipse, Arduino (IDE), entre otros.

Arduino (IDE). Según Arduino (2019) el software de código abierto Arduino (IDE) facilita escribir código y cargarlo en la placa del arduino seleccionado. Se ejecuta en Windows, Mac OS X y Linux. El entorno está escrito en Java, se basa en el lenguaje Processing y la plataforma Wiring (software de código abierto) que utiliza la sintaxis simplificada de C/C++. El Arduino (ID) de acuerdo a Artero (2013) permite escribir y editar el programa (también llamado “sketch” en el mundo de Arduino), comprobar que no haya ningún error y grabarlo en la memoria del microcontrolador de la placa Arduino para que este se convierta a partir de entonces en el ejecutor autónomo de dicho programa.

Sistemas Operativos. Teniendo en cuenta a del Prado y Lamas (2013), los sistemas operativos pueden ser clasificados de la siguiente forma:

Multiusuario: Permite que dos o más usuarios utilicen sus programas al mismo tiempo. Algunos sistemas operativos permiten a centenares o millares de usuarios al mismo tiempo.

Multiprocesador: soporta el abrir un mismo programa en más de una CPU.

Multitarea: Permite que varios programas se ejecuten al mismo tiempo.

Multitramo: Permite que diversas partes de un solo programa funcionen al mismo tiempo.

Tiempo Real: Responde a las entradas inmediatamente. Los sistemas operativos como DOS y UNIX, no funcionan en tiempo real.

Los sistemas operativos proporcionan una plataforma de software encima de la cual otros programas, llamados aplicaciones, puedan funcionar. Las aplicaciones se programan para que funcionen encima de un sistema operativo particular, por tanto, la elección del sistema operativo determina en gran medida las aplicaciones que puedes utilizar.

Algunos sistemas operativos son:

DOS: Familia de sistemas operativos para PC. Sus siglas significan Disk Operating System. Fue creado para ordenadores IBM y fue muy popular. Carece de interfaz gráfica y no es multiusuario ni multitarea. La última versión MS-DOS 8.0, con la aparición del sistema operativo Windows fue rápidamente sustituido.

WINDOWS: Familia de sistemas operativos no libres desarrollados por la empresa Microsoft Corporation, que se basan en una interfaz gráfica que se caracteriza por la utilización de ventanas. La última versión es Windows 10.

MACOS: Familia de sistemas operativos no libres desarrollados creado por Apple para su línea de computadoras Macintosh. La última versión High Sierra (versión 10.13.1)

UNÍX: Familia de sistemas operativos que comparten unos criterios de diseño e interoperabilidad en común, que descienden de una primera implementación original de AT&T. Se trata de un sistema operativo portable, multitarea y multiusuario. La última versión UNIX Versión 7.

GNU/LINUX: Sistema operativo libre creado por Richard Stallman. GNU es un acrónimo que significa GNU no es Unix («GNU's Not Unix»). La última versión varía según las distribuciones existentes: Debian, Ubuntu, Linux Mint, OpenWRT, entre otros. Un sistema operativo libre quiere decir que los códigos completos del sistema estarán disponibles para todo el mundo, sin tener que pagar por un programa. Como resultado, un usuario que necesita cambios en el sistema será siempre libre para hacerlos por sí mismo, o de contratar a cualquier programador o empresa disponible para hacerlos por él. Los usuarios no estarán ya a merced de un programador o una empresa que sea dueña de los códigos fuente y sea la única en posición de hacer cambios. Según su creador, R. Stallman, un software es libre si cumple estas condiciones: Cualquiera tiene libertad para ejecutar el programa, con cualquier propósito, Cualquiera tiene libertad para modificar el programa para adaptarlo a sus necesidades y para ello, se debe tener acceso al código fuente, porque modificar un programa sin disponer del código fuente es

extraordinariamente difícil. Se tiene la libertad para redistribuir copias, tanto gratis como por un canon, se tiene la libertad para distribuir versiones modificadas del programa, de tal manera que la comunidad pueda beneficiarse con sus mejoras.

Sistema Operativo OpenWrt. El proyecto OpenWrt es un sistema operativo Linux dirigido a dispositivos integrados. En lugar de intentar crear un firmware único y estático, OpenWrt proporciona un sistema de archivos completamente grabable con administración de paquetes. Esto lo libera de la selección y configuración de la aplicación proporcionada por el proveedor y le permite personalizar el dispositivo mediante el uso de paquetes para adaptarse a cualquier aplicación. Para los desarrolladores, OpenWrt es el marco para construir una aplicación sin tener que construir un firmware completo a su alrededor; para los usuarios, esto significa la capacidad de personalización completa, para utilizar el dispositivo de una manera nunca prevista. OpenWrt se ofrece de forma gratuita a través de su licencia GPL, y gracias a los esfuerzos de muchos colaboradores voluntarios (tanto individuos como empresas). No hay cuotas de suscripción o licencia. (Endt *et al.*, 2019)

Base de Datos.

Definición. Según Marqués (2011) una base de datos es un conjunto de datos almacenados en memoria externa que están organizados mediante una estructura de datos. Cada base de datos ha sido diseñada para satisfacer los requisitos de información de una empresa u otro tipo de organización, como por ejemplo, una universidad o un hospital.. Una base de datos se puede percibir como un gran almacén de datos que se define y se crea una sola vez, y que se utiliza al mismo tiempo por distintos usuarios. En una base de datos todos los datos se integran con una mínima cantidad de duplicidad. De este modo, la base de datos no pertenece a un solo departamento sino que se comparte por toda la organización. Además, la base de datos no sólo contiene los datos de la organización, también almacena una descripción de dichos datos. Esta descripción es lo que se denomina metadatos, se almacena en el diccionario de datos o catálogo y es lo que permite que exista independencia de datos lógica-física.

Sistema de gestión de bases de datos. De acuerdo a Marqués (2011) el sistema de gestión de la base de datos (en adelante SGBD) es una aplicación que permite a los usuarios definir, crear y mantener la base de datos, además de proporcionar un acceso controlado a la misma. Se denomina sistema de bases de datos al conjunto formado por la base de datos, el SGBD y los programas de aplicación que dan servicio a la empresa u organización.

Generalmente, un SGBD proporciona los servicios que se citan a continuación:

- a) El SGBD permite la definición de la base de datos mediante un lenguaje de definición de datos. Este lenguaje permite especificar la estructura y el tipo de los datos, así como las restricciones sobre los datos.

b) El SGBD permite la inserción, actualización, eliminación y consulta de datos mediante un lenguaje de manejo de datos. El hecho de disponer de un lenguaje para realizar consultas reduce el problema de los sistemas de ficheros, en los que el usuario tiene que trabajar con un conjunto fijo de consultas, o bien, dispone de un gran número de programas de aplicación costosos de gestionar. Hay dos tipos de lenguajes de manejo de datos: los procedurales y los no procedurales. Estos dos tipos se distinguen por el modo en que acceden a los datos. Los lenguajes procedurales manipulan la base de datos registro a registro, mientras que los no procedurales operan sobre conjuntos de registros. En los lenguajes procedurales se especifica qué operaciones se debe realizar para obtener los datos resultado, mientras que en los lenguajes no procedurales se especifica qué datos deben obtenerse sin decir cómo hacerlo. El lenguaje procedural más utilizado es el SQL (Structured Query Language) que, de hecho, es un estándar y es el lenguaje de los SGBD relacionales.

c) El SGBD proporciona un acceso controlado a la base de datos mediante:

Un sistema de seguridad, de modo que los usuarios no autorizados no puedan acceder a la base de datos.

- Un sistema de integridad que mantiene la integridad y la consistencia de los datos.
- Un sistema de control de concurrencia que permite el acceso com-partido a la base de datos.
- Un sistema de control de recuperación que restablece la base de datos después de que se produzca un fallo del hardware o del software.
- Un diccionario de datos o catálogo, accesible por el usuario, que contiene la descripción de los datos de la base de datos.

Lenguaje SQL. De acuerdo a Marqués (2011) las siglas SQL corresponden a Structured Query Language (lenguaje de consulta estructurado), un lenguaje estándar que permite manejar los datos de una base de datos relacional. La mayor parte de los SGBD relacionales implementan este lenguaje y mediante él se realizan todo tipo de accesos a la base de datos.

Normalmente, cuando un SGBD relacional implementa el lenguaje SQL, todas las acciones que se pueden llevar a cabo sobre el sistema se realizan mediante sentencias de este lenguaje. Dentro de SQL hay varios tipos de sentencias que se agrupan en tres conjuntos:

- Sentencias de definición de datos: son las sentencias que permiten crear tablas, alterar su definición y eliminarlas. En una base de datos relacional existen otros tipos de objetos además de las tablas, como las vistas, los índices y los disparadores. Las sentencias para crear, alterar y eliminar vistas e índices también pertenecen a este conjunto.
- Sentencias de manejo de datos: son las sentencias que permiten insertar datos en las tablas, consultarlos, modificarlos y borrarlos.
- Sentencias de control: son las sentencias que utilizan los administradores de la base de datos para realizar sus tareas, como por ejemplo crear usuarios y concederles o revocarles privilegios.

Las sentencias de SQL se pueden escribir tanto en mayúsculas como en minúsculas, y lo mismo sucede con los nombres de las tablas y de las columnas. Para facilitar la lectura de

los ejemplos, se utilizará mayúsculas para las palabras clave del lenguaje y minúsculas para los nombres de tablas y de columnas. En los ejemplos se introducirán espacios en blanco para tabular las expresiones. Las sentencias de SQL terminan siempre con el carácter punto y coma (;).

Sqlite. * Historia: cuando D. Richard Hipp trabajaba desarrollando Software para la fuerza naval de los Estados Unidos, comenzó a desarrollar SQLite, según él por una necesidad personal para su uso personal. En Enero de 2000 D. Richard Hipp estaba trabajando con su equipo de la General Dynamics en la Fuerza Naval de los Estados Unidos, en un proyecto de software, el cual se conectaba a una base de datos Informix, el motor funcionaba muy bien, pero habían tenido problemas para hacer una reconfiguración cuando el sistema se reiniciaba. Luego cambiaron a PostgreSQL, pero administrar la base de datos era un poco más complejo. Fue en ese momento cuando surgió la idea de escribir un simple motor de base de datos SQL que permitiera leer los archivos del disco duro, y luego ser llamados en diferentes solicitudes.

* Definición: según Consortium (2019) SQLite es una biblioteca en proceso que implementa un motor de base de datos transaccional de SQL autocontenido, sin servidor, de configuración cero. El código para SQLite está en el dominio público y, por lo tanto, es de uso gratuito para cualquier propósito, comercial o privado. SQLite es la base de datos más implementada en el mundo con más aplicaciones de las que podemos contar, incluidos varios proyectos de alto perfil. SQLite es un motor de base de datos SQL incorporado. A diferencia de la mayoría de las otras bases de datos SQL, SQLite no tiene un proceso de servidor separado. SQLite lee y escribe directamente en archivos de disco ordinarios. Una completa base de datos SQL con múltiples tablas, índices, disparadores y vistas, está contenida en un solo archivo de disco. El formato de archivo de la base de datos es multiplataforma: puede copiar libremente una base de datos entre sistemas de 32 y 64 bits o entre arquitecturas big-endian y little-endian . Estas características hacen de SQLite una opción popular como formato de archivo de aplicación. Los archivos de base de datos SQLite son un formato de almacenamiento recomendado por la Biblioteca del Congreso de los Estados Unidos. SQLite es una biblioteca compacta. Con todas las funciones habilitadas, el tamaño de la biblioteca puede ser inferior a 600 KB, según la plataforma de destino y la configuración de optimización del compilador. SQLite generalmente se ejecuta más rápido cuanto más memoria le das. Sin embargo, el rendimiento suele ser bastante bueno incluso en entornos con poca memoria. Dependiendo de cómo se utilice, SQLite puede ser más rápido que la E / S directa del sistema de archivos. SQLite se prueba cuidadosamente antes de cada lanzamiento y tiene una reputación de ser muy confiable. La mayor parte del código fuente de SQLite está dedicado exclusivamente a pruebas y verificación. Un conjunto de pruebas automatizadas ejecuta millones y millones de casos de prueba que involucran cientos de millones de declaraciones SQL individuales y logra una cobertura de prueba de rama del 100%. SQLite responde con gracia a los fallos de asignación de memoria y los errores de E/S del disco. Las transacciones son ACID incluso si son interrumpidas por fallas del sistema o fallas de

energía. Todo esto se verifica mediante las pruebas automatizadas utilizando arneses de prueba especiales que simulan fallas del sistema. Por supuesto, incluso con todas estas pruebas, todavía hay errores. Pero a diferencia de algunos proyectos similares (especialmente los competidores comerciales), SQLite es abierto y honesto con respecto a todos los errores y proporciona listas de errores y cronologías minuto a minuto de los cambios de código.

La base de código SQLite es compatible con un equipo internacional de desarrolladores que trabajan en SQLite a tiempo completo. Los desarrolladores continúan expandiendo las capacidades de SQLite y mejoran su confiabilidad y rendimiento, al tiempo que mantienen la compatibilidad hacia atrás con la especificación de interfaz publicada , la sintaxis SQL y el formato de archivo de base de datos. El código fuente es absolutamente gratuito para cualquiera que lo desee, pero también hay disponible asistencia profesional .

El proyecto SQLite se inició el 09/05/2000. El futuro siempre es difícil de predecir, pero la intención de los desarrolladores es apoyar SQLite hasta el año 2050. Las decisiones de diseño se toman con ese objetivo en mente.

* Características de la base de datos

De acuerdo a Montiel (2008)

- Fichero único. La base de datos se almacena en un único fichero, cuyo formato es multiplataforma (Es posible leer el fichero en sistemas de 32 y 64 bits o en arquitecturas big-endian y little-endian). Estas características hacen que SQLite sea popular para usarlo como formato de archivo de las aplicaciones. O dicho de otra forma: Se puede usar SQLite como sustituto de Oracle o como sustituto de fopen().

- Manifiesto de tipado. La mayoría de motores SQL utilizan tipado estático. Cada columna de una tabla se asocia con un tipo de datos, y solo pueden introducirse valores de un tipo particular. SQLite elimina esta restricción, y hace que el tipo de datos pueda ser una propiedad del valor en sí, y no de la columna.

- Registros de longitud variable. Muchos otros motores SQL, fijan una cantidad de espacio para cada todas las filas, de forma que, por ejemplo, si declaramos una columna como varchar(100), el motor reservará 100 bytes de espacio para todas las filas sin tener en cuenta la información que se guarde. SQLite, por el contrario, usa sólo la cantidad de disco que necesita para almacenar la información en una fila.

- Seguridad de los datos. Más de dos tercios del código están dedicados puramente a la prueba y verificación. Una aplicación automatizada ejecuta cientos de miles de pruebas empleando millones de consultas SQL. SQLite responde perfectamente a fallos de reserva de memoria, y errores de E/S de disco.

* Características del gestor de la base de datos

De acuerdo a Montiel (2008)

- Embebido: La mayoría de motores de bases de datos, como MySQL, Oracle o SQL Server, están implementados como un servicio (o demonio en Unix). Los programas que quieren acceder a la base de datos se comunican con el servidor usando algún tipo de protocolo

para enviar peticiones y recibir resultados. Esto es lo que se conoce como una aplicación cliente-servidor. SQLite no funciona de esta manera, con SQLite, el proceso que quiere acceder a la base de datos, lee y escribe directamente en disco, no hay servicio intermediario, de esta manera se puede hacer una aplicación totalmente autónoma y portable. Esto tiene ventajas y desventajas. La principal ventaja es que no debemos tener un servicio que instalar, configurar, inicializar, mantener, etc. Por otro lado, el uso de un servidor para la base de datos provee mayor protección frente a bugs en el lado de cliente. Un fallo de segmentación en el cliente no puede afectar a la memoria que se encuentra en el servidor. Además, como el servidor es un único proceso, puede controlar mejor la concurrencia. No obstante, una característica de SQLite es que es la única base de datos sin servidor (que el autor sepa) que permite el acceso de múltiples aplicaciones a la misma base de datos.

- No necesita configuración: Debido a que SQLite es un SGBD embebido en la aplicación, no necesita instalar ni configurar nada más aparte de la aplicación en cuestión.

- Transaccional: Una base de datos transaccional es aquella cuyos cambios y consultas son atómicos, consistentes, aislados y durables (ACID), y por tanto es capaz de realizar transacciones seguras. Las transacciones en SQLite tienen estas características, incluso cuando se interrumpen por el fallo del programa, del sistema operativo o de la alimentación del ordenador. Todos los cambios de una transacción en SQLite se hacen completamente o no se hacen.

* Casos en los que es adecuado el uso de SQLite

De acuerdo a Montiel (2008)

- En aplicaciones autónomas que no queramos que dependan de otros procesos, como podría ser un programa para llevar la contabilidad de una pequeña/mediana empresa. – Para su uso como formato de almacenamiento para cualquier aplicación.

- En pequeños dispositivos que no disponen de servicio SQL (PDAs, Móviles, etc.) y con pocos recursos de memoria.

- Bases de datos internas de programas que necesiten manejar gran cantidad de información temporal (p. ej. videojuegos, programas de análisis, etc.).

Redes.

Definiciones. Redes de computadora: Según Tanenbaum y Wetherall (2012) la fusión de las computadoras y las comunicaciones ha tenido una profunda influencia en cuanto a la manera en que se organizan los sistemas de cómputo, el concepto una vez dominante del “centro de cómputo” como un salón con una gran computadora a la que los usuarios llevaban su trabajo para procesarlo es ahora totalmente obsoleto (aunque los centros de datos que contienen miles de servidores de Internet se están volviendo comunes), el viejo modelo de una sola computadora para atender todas las necesidades computacionales de la organización se ha reemplazado por uno en el que un gran número de computadoras separadas pero interconectadas realizan el

trabajo. A estos sistemas se les conoce como redes de computadoras.

Según del Prado y Lamas (2013) es un conjunto de dos o más computadores interconectadas entre sí y que intercambian información, los objetivos de las redes consisten en «compartir recursos», y uno de sus objetivos es hacer que todos los programas, datos y equipo estén disponibles para cualquiera de la red que así lo solicite, sin importar la localización física del recurso y del usuario.

Tecnología de redes: Según Olifer y Olifer (2009) es un conjunto coordinado de software y hardware (por ejemplo, controladores, adaptadores de red, cables y conectores) y mecanismos para la transmisión de datos a través de enlaces de comunicación, suficientes para construir una red de computadoras.

Protocolo: Según Kurose y Ross (2010) un protocolo define el formato y el orden de los mensajes intercambiados entre dos o más entidades que se comunican, así como las acciones tomadas en la transmisión y/o la recepción de un mensaje u otro suceso. Internet, y las redes de computadoras en general, hacen un uso extensivo de los protocolos.

Tipos de redes. PAN (Personal Area Network): Según Tanenbaum y Wetherall (2012) las redes de área personal permiten a los dispositivos comunicarse dentro del rango de una persona. Un ejemplo común es una red inalámbrica que conecta a una computadora con sus periféricos, como ejemplo las redes Bluetooth para conectar estos componentes sin necesidad de cables, utilizan el paradigma maestro-esclavo, la unidad del sistema (la PC), por lo general es el maestro que trata con el ratón, el teclado, etc., como sus esclavos. El maestro dice a los esclavos qué direcciones usar, cuándo pueden transmitir información, durante cuánto tiempo pueden transmitir, qué frecuencias usar, etcétera. Las redes PAN también se pueden construir con otras tecnologías que se comunican dentro de rangos cortos, como RFID en las tarjetas inteligentes y los libros de las bibliotecas.

LAN (Local Area Network): Según del Prado y Lamas (2013) la red de área local surgió de la necesidad de compartir de manera eficaz datos y servicios entre usuarios de una misma área de trabajo, son redes que cubren pequeñas áreas geográficas tales como un cuarto o un edificio, esta extensión suele ser inferior a los cinco kilómetros.

MAN (Metropolitan Area Network): Según Tanenbaum y Wetherall (2012) una red de área metropolitana cubre toda una ciudad, cabe mencionar que la televisión por cable no es la única MAN, los recientes desarrollos en el acceso inalámbrico a Internet de alta velocidad han originado otra, la cual se estandarizó como IEEE 802.16 y se conoce comúnmente como WiMAX.

WAN (Wide Area Network): Según del Prado y Lamas (2013) la red de área extensa son un conjunto de redes locales interconectadas dentro de un área metropolitana. Las líneas utilizadas para realizar esta interconexión suelen ser parte de las redes públicas de transmisión de datos.

Tabla 8: Clasificación de los procesadores interconectados con base en la escala. Fuente: Tanenbaum y Wetherall (2012)

Distancia entre procesadores	Procesadores ubicados en el (la) mismo(a)	Ejemplo
1 m	Metro cuadrado	Red de área personal
10 m	Cuarto	
100 m	Edificio	Red de área local
1 km	Campus	
10 km	Ciudad	Red de área metropolitana
100 km	País	
1000 km	Continente	Red de área amplia
10000 km	Planeta	Internet

Topología de red. Según Olifer y Olifer (2009) la topología de red se refiere a la configuración de una gráfica cuyos vértices corresponden a los nodos de red (por ejemplo, computadoras) y al equipo de comunicaciones (por ejemplo, ruteadores), cuyos extremos representan las conexiones físicas o de información entre ellos.

De acuerdo a del Prado y Lamas (2013) el término topología se emplea para referirse a la disposición geométrica de las estaciones de una red y los cables que las conectan, y al trayecto seguido por las señales a través de la conexión física. La topología de la red es pues, la disposición de los diferentes componentes de una red y la forma que adopta el flujo de información. Para determinar qué topología resulta más adecuada para una red concreta se tienen en cuenta numerosos parámetros y variables, como el número de máquinas que se van a interconectar, el tipo de acceso al medio físico deseado, etc.

Según Dordogne (2015) se distingue entre la topología física, en relación con el plano de la red, y la topología lógica, que identifica la forma en que la información circula por el nivel más bajo. La interconexión entre nodos de la red se realiza en forma de conexión punto a punto, es decir, uno a uno, o en forma de multipuntos, n nodos con n nodos.

Medios físicos de conexión de red. Según Kurose y Ross (2010) los medios físicos se pueden clasificar dentro de dos categorías:

Guiados: en los medios guiados, las ondas se transportan a través de un medio sólido, como por ejemplo un cable de fibra óptica, un cable de cobre de par trenzado o un cable coaxial.

No Guiado: en los medios no guiados, las ondas se propagan por la atmósfera y el espacio exterior, tal como ocurre en las redes LAN inalámbricas o en un canal de satélite digital.

Topología física de redes. Teniendo en cuenta a Olifer y Olifer (2009) las topologías físicas de red son:

- *Malla*: se obtiene a partir de la topología totalmente conectada al suprimir algunos de sus enlaces, hacen posible que un gran número de computadoras se puedan conectar y es una característica típica de las redes de gran tamaño.

- *Anillo*: los datos se transmiten alrededor del anillo de computadora a computadora. La ventaja principal del anillo consiste en su propiedad para proporcionar enlaces redundantes, cada par de nodos se conecta mediante dos rutas: una en el sentido de las manecillas del reloj y la otra en el sentido opuesto, el anillo representa una configuración muy apropiada para brindar retroalimentación, pues los datos, una vez que han recorrido todo el anillo, regresan a su nodo de origen, debido a esta característica, el nodo de origen puede controlar el proceso de entrega de los datos al nodo de destino, con mucha frecuencia, esta propiedad del anillo se utiliza para probar la conectividad de la red y para buscar aquellos nodos que no funcionan de manera correcta. Por otro lado, en las redes con topología de anillo es necesario tomar medidas especiales para asegurarse de que, cuando una computadora falle o deje de operar de forma temporal, no falle el circuito de comunicaciones de los demás nodos de la red.

- *Estrella*: supone que cada computadora está conectada directamente a un dispositivo central llamado concentrador. Las funciones del concentrador incluyen el redireccionamiento de la información de una computadora a una computadora específica o a todas las computadoras que forman la red. Como concentrador, es factible utilizar computadoras universales o dispositivos de red especializados. Esta topología de red tiene sus desventajas, por ejemplo: el alto costo del equipo de la red debido a la necesidad de adquirir dispositivos especializados para el nodo central.

- *Árbol*: además, las posibilidades de incrementar el número de nodos en la red están limitadas por el número de puertos disponibles en el concentrador, a veces tiene sentido construir la red utilizando varios concentradores conectados entre sí en forma jerárquica mediante enlaces del tipo estrella, la estructura resultante también se conoce como estrella jerárquica o árbol. En la actualidad, la estructura en árbol es la más común y se usa ampliamente tanto en LAN como en WAN.

- *Bus Común*: es una versión especial de la topología en estrella, en este caso, el papel que desempeña el elemento central se asigna al cable pasivo al cual se encuentran conectadas varias computadoras de acuerdo con el diseño de una compuerta OR alambrada. La mayoría de las redes inalámbricas tienen la misma topología; sin embargo, en este caso el medio común de transmisión por radio desempeña el papel del bus común. La información se transmite a través del cable y se encuentra simultáneamente disponible en todas las computadoras conectadas a este cable. Las principales ventajas de este diseño son su bajo costo y su simplicidad para conectar nuevos nodos a la red. La desventaja más significativa de bus común es su baja confiabilidad, ya que cualquier defecto en el cable o en cualquiera de los conectores paraliza toda la red. Otra desventaja del bus común es su bajo desempeño, pues este método de conexión significa que solamente una computadora puede transmitir datos a través de la red a la vez. Por tanto, el ancho de banda del enlace de comunicación siempre se divide entre todos los nodos

de la red. Hasta fechas recientes, el bus común fue una de las topologías más populares de las LAN.

- *Híbrida*: Mientras que las redes pequeñas, en general, tienen una de las topologías típicas estrella, anillo o bus común, las redes de gran tamaño están caracterizadas por la presencia de conexiones arbitrarias entre computadoras. En dichas redes, uno puede distinguir fragmentos conectados de forma arbitraria (subredes) que cuentan con una topología típica. Por tanto, dichas redes se conocen como redes de topología híbrida.

Métodos de acceso al soporte. Teniendo en cuenta a Dordogne (2015) En un canal punto a punto, un emisor puede transmitir libremente. En cambio, cuando el soporte es compartido por varios periféricos, es necesario administrar la forma en que se intercambian datos. Estos intercambios dependen de la arquitectura de red, es decir, de la topología lógica. Un modo de acceso define las normas que regulan el acceso a cada componente, la transmisión y la liberación del canal compartido. Se distinguen esencialmente tres modos: el de contención, el polling y el paso de testigo.

a. *La contención*: En el modo de acceso por contención, cada equipo emite cuando lo necesita, después de escuchar el canal o el portador (CS - Carrier Sense), que debe estar disponible. Se escucha la trama emitida para comprobar que ninguna señal perturba la emisión. Es este caso no hay ningún arbitraje del canal. Dos equipos pueden emitir simultáneamente, lo que puede conducir a una sobre tensión en el caso de un cable coaxial, o a una recepción de información sobre el par receptor mientras se emiten algunos datos sobre el otro par (par trenzado). Este estado se denomina de colisión, y es necesario detectarlo. Estas situaciones ocurren si respetamos algunas de las condiciones (por ejemplo, el alcance máximo de la red). Por ello, con Ethernet, el tamaño de una trama no debe ser inferior a 64 bytes, de manera que el equipo emisor sea capaz de detectar la colisión antes de haber enviado el último byte de la trama. Cuando se produce una colisión, el primer equipo que lo detecta prolonga su emisión a través de una señal especial (trama de interferencia o JAM), con el fin de avisar a los otros equipos de que hubo una colisión. En este caso, se define aleatoriamente un tiempo de espera para cada equipo que emitía en el momento de la colisión. Así no intentarán reanudar el control del canal en el mismo momento. Las dos implementaciones más extendidas de la contención son CSMA/CD y CSMA/CA. CSMA corresponde a la escucha del portador (Carrier Sense) en un soporte compartido (Multiple Access). Las dos implementaciones se distinguen por el hecho de que una detecta las colisiones (Collision Detection) y la otra intenta evitarlas (Collision Avoidance). El segundo caso es una alternativa con relación al modo anteriormente mencionado. De hecho, en vez de intentar transmitir los datos arriesgándose a una colisión (después de escuchar el soporte), el dispositivo envía una trama preliminar para informar a los otros equipos de que quiere utilizar el canal (para enviar su trama de datos). CSMA/CD corresponde a la implementación Ethernet, mientras que CSMA/CA es la adoptada por la norma 802.11 (Wi-Fi).

b. *La pregunta (polling)*: Con este método se designa a un equipo de la red como

administrador del acceso al canal. Este hardware, que es el supervisor, pregunta en un orden predeterminado a cada uno de los nodos si tienen información para transmitir. A menudo, el supervisor es un hub o switch y el hardware secundario son los nodos de la estrella. La ventaja de este método es que se centralizan todos los accesos al canal. Además, el tiempo de acceso y el volumen de datos manejado en este soporte son previsibles y fijos. Sin embargo, utiliza una parte del ancho de banda de la red para emitir mensajes de gestión (preguntas, advertencias, entregas...), lo que implica un mayor coste de ancho de banda.

c. *El paso de testigo*: En el método del paso de testigo, las tramas van pasando de equipo en equipo, y cada uno de estos se comporta como un repetidor. Inicialmente, una pequeña trama, el testigo (token), se repite de equipo en equipo hasta que una máquina que desee emitir lo conserva durante un tiempo determinado. Este método de acceso se utiliza generalmente en arquitecturas de anillo (ring). Este testigo es, por decirlo de algún modo, un mensaje de autorización que ofrece la exclusividad del soporte a la estación que lo posee. Esta emite una trama que se repetirá en cada estación hasta dar la vuelta al anillo. A su paso, el destinatario de la trama, que ve pasar la señal, realiza una copia, si no es errónea y si dispone de suficiente espacio en el buffer de recepción. El destinatario marca la trama copiada con el fin de informar al emisor de que se ha leído. Después de haber dado la vuelta al anillo, el emisor retira la trama, que libera al testigo dejándolo disponible para el próximo equipo. El paso de testigo aplica una solución determinista que permite un buen control del soporte. La velocidad máxima real alcanzada es mucho más elevada que con el método de contención, propenso a las colisiones. Hay varias normas de paso de testigo para las topologías de anillo (IEEE 802.5, Token Ring o FDDI), pero también para las topologías de bus (IEEE 802.4). Todavía se encuentran redes que implementan Token Ring o FDDI.

Componentes de una red. Según del Prado y Lamas (2013) una red básica está compuesta por:

- *Servidor*: este ejecuta el sistema operativo de red y ofrece los servicios de red a las estaciones de trabajo.

- *Estaciones de Trabajo*: Cuando una computadora se conecta a una red, la primera se convierte en un nodo de la última y se puede tratar como una estación de trabajo o cliente. Las estaciones de trabajos pueden ser computadoras personales con el DOS, Macintosh, Unix, OS/2 o estaciones de trabajos sin discos.

- *Tarjetas o Placas de Interfaz de Red*: Toda computadora que se conecta a una red necesita de una tarjeta de interfaz de red que soporte un esquema de red específico, como Ethernet, ArcNet o Token Ring. El cable de red se conectar a la parte trasera de la tarjeta.

- *Sistema de Cableado*: El sistema de la red está constituido por el cable utilizado para conectar entre si el servidor y las estaciones de trabajo.

- *Recursos y Periféricos Compartidos*: Entre los recursos compartidos se incluyen los dispositivos de almacenamiento ligados al servidor, las unidades de discos ópticos, las

impresoras, los trazadores y el resto de equipos que puedan ser utilizados por cualquiera en la red.

IEEE 802. Según Dordogne (2015) el proyecto 802 del Institute of Electrical and Electronic Engineers (IEEE) divide la capa de enlace de datos (OSI) en dos sub-capas inferiores. La primera se llama control de acceso a los medios de comunicación o Medium Access Control (MAC) y es inherente a cada tipo de red. La segunda se llama control de la conexión lógica o Logical Link Control (LLC), es independiente del tipo de red y está estandarizada como 802.2.

Tabla 9: Las normas 802.x del IEEE. Fuente: Dordogne (2015)

Conexión	IEEE 802.2 (LLC)		
	IEEE 802.3	IEEE 802.4	IEEE 802.5
Física	CSMA/CD	Token Bus	Token Ring

IEEE 802.1 (gestión y interconexión de redes). Según Dordogne (2015) esta especificación es una introducción a las normas 802 e implica las relaciones globales entre todas las otras especificaciones. Tiene en cuenta los problemas relativos a la gestión de los sistemas y a la interconexión de redes. Por ejemplo, el algoritmo de spanning tree, que soluciona los problemas de bucles en las redes Ethernet, se describe en 802.1D. El marcado en las VLAN explícitas se define en 802.1Q. La gestión de las prioridades se aborda en 802.1P.

IEEE 802.2 (LLC). Teniendo en cuenta a Dordogne (2015) tenemos:

- *Principios de Logical Link Control (LLC):* El control de capa de conexión define servicios estándar, cualquiera que sea la topología y el método de acceso al soporte. Controlan conexiones punto a punto o multipunto en soportes limitados o ilimitados, en half-duplex o en full-duplex, en redes de conmutación de paquetes o de circuitos. LLC puede garantizar la integridad de la transmisión de extremo a extremo entre dos estaciones. High level Data Link Control (HDLC) es un protocolo creado en 1979 que corresponde a una etapa intermedia de LLC.

- *Tipos de servicio:* Los protocolos LLC ofrecen tres tipos de servicio.

* Servicio sin conexión ni confirmación (tipo 1): Es el servicio más simple y menos fiable. Es también el más utilizado, ya que a menudo la mayoría de los protocolos utilizan un transporte seguro (capa 4 OSI).

* Servicio orientado a la conexión (tipo 2): Consta de una conexión lógica entre el emisor y el receptor.

* Servicio sin conexión con confirmación (tipo 3): Cada trama se confirma individualmente, pero no existe conexión lógica entre las dos estaciones.

IEEE 802.3 y Ethernet. Teniendo en cuenta a Dordogne (2015) el protocolo Ethernet aparece en 1980 como resultado de los trabajos de DEC, Intel y Xerox. Como es tan económico, está siempre presente en cualquier instalación. Su evolución se estandarizó en la norma IEEE 802.3, que cubre la capa Física y una parte de la de Conexión de datos. Hay algunas diferencias entre Ethernet y 802.3, pero no disociaremos estas dos normas. IEEE 802.3 utiliza los servicios de la capa LLC. El modo de acceso al soporte es por contención, a través de Carrier Sense Multiple Access/Collision Detection (CSMA/CD).

- *Características de la capa Física:* IEEE 802.3 ofrece distintas opciones de capa Física. Las denominaciones utilizadas toman en cuenta la velocidad, el soporte, el tipo de señal y la longitud del segmento. 10 base 2 corresponde a una red Ethernet con cable coaxial fino (10 Mbps, señalización digital para segmentos de 200 metros como máximo; en realidad, 185 metros). 10 base T corresponde a una Ethernet con par trenzado (T por Twisted pair). 100 base T es una implementación de Ethernet a 100 Mbps con par trenzado (categoría 5). 1000 base T (norma IEEE 802.3ab) corresponde a una solución basada en el par trenzado UTP categoría 5e (máximo 100 metros). 10 G base T se basa en la definición IEEE 802.3an y establece la velocidad a 10 Gbps para el par trenzado en longitudes máximas de 100 metros. La señalización es digital y la codificación utilizada es la Manchester (las tarjetas de red funcionan a 20 MHz). Una trama tiene una longitud mínima de 64 bytes y máxima de 1518 bytes.

IEEE 802.5 y Token Ring. Teniendo en cuenta a Dordogne (2015) el protocolo de red cableada local Token Ring ha estado respaldado mucho tiempo por su fabricante, IBM. Su competidor, Ethernet, menos costoso y fácil de utilizar, aunque menos eficaz, ha acabado por reemplazarlo. Aún podemos encontrar esta red, estandarizada como 802.5 por IEEE, en infraestructuras importantes y antiguas. Pero cada vez es menos frecuente.

- *Configuración de la red:* La norma 802.5 se basa en el método del paso de testigo en una topología de anillo. Hay distintas combinaciones posibles para los anchos de banda, 4 Mbps, 16 Mbps y 100 Mbps, en función del cableado utilizado, UTP, STP o fibra multimodo. Los equipos se conectan en estrella (topología física) al Multistation Access Unit (MAU). Los MAU más antiguos disponen de tomas macho y hembra y se utilizan conectores DB9 para conectar las tarjetas a los MAU, a través de un cable Token Ring específico. La topología lógica es en anillo punto a punto, donde cada máquina actúa como repetidor.

- *Autorreconfiguración del anillo:* Un punto importante en Token Ring es la gestión del anillo mediante un equipo dedicado (aquel que lleve más tiempo conectado ya que este hecho le proporciona la dirección MAC más alta): el supervisor activo. Este equipo se encarga de solucionar una serie de problemas, como; la detección de la pérdida del testigo (cuando inesperadamente desaparece la estación que posee el testigo), la eliminación de una trama que completó su ciclo (el emisor desaparece antes de eliminar su trama del anillo) y la detección de los errores en los mecanismos de prioridad (un testigo no debe generarse con la misma prioridad dos veces seguidas, salvo si su prioridad es la más baja).

IEEE 802.11 y Wi-Fi. Teniendo en cuenta a Dordogne (2015) las conexiones inalámbricas de tipo Radio LAN (RLAN) encontraron estándares adecuados gracias a los trabajos de los grupos IEEE 802.11.

- *Presentación.*

En 1997 el grupo de trabajo 802.11 estandariza, después de varios años de trabajo, la definición de redes de tipo Wireless LAN, que se retocó en 1999. Como en el caso de 802.3, estas especificaciones cubren las capas Física y Conexión de datos del modelo OSI. Esta última está dividida en dos subcapas: Medium Access Control (MAC), para el acceso al soporte de transmisión, y Logical Link Control (LLC), para el control de la transmisión. En la capa Física, 802.11 define tres modos de transmisión. El primero está basado en la difusión infrarroja, que finalmente no se utilizará en las implementaciones de estas especificaciones. Las otras dos tecnologías utilizan la transmisión por radio. Finalmente, una sola, denominada Direct Sequence Spread Spectrum (DSSS), se implementará. Las especificaciones 802.11 interesan tanto a algunos fabricantes que en 1999 forman la asociación Wireless Ethernet Compatibility Alliance (WECA). Su objetivo no es solamente promover este nuevo estándar, sino también certificar dispositivos con el fin de garantizar su buen funcionamiento. El certificado Wireless Fidelity (Wi-Fi), patente de interoperabilidad, se otorga después de algunas pruebas. Finalmente, este organismo se rebautiza como Wi-Fi Alliance. Después de las pruebas de compatibilidad, el fabricante del hardware puede etiquetar las cajas con el siguiente logotipo, siempre que respete los estándares exigidos.

Figura 22: Logo Wi-Fi. Fuente: Dordogne (2015)



El estándar Wi-Fi permite la conectividad a distancias que superan algunas decenas de metros. La utilización de antenas permite alcanzar varios centenares de metros.

- *Normas de la capa Física.*

Hay muchas especificaciones que se han basado en la 802.11 original, de las cuales tres definen el uso de la capa física.

* 802.11b: Esta norma, publicada en septiembre de 1999, aumenta la velocidad máxima de transmisión a 11 Mbps, con velocidades que pueden llegar a 5,5, 2 y 1 Mbps. La frecuencia de trabajo es de 2,4 GHz. A partir de ahora, la red tiene un nombre, el SSID (Service Set Identifier o identificador de paquetes de servicio).

* 802.11a: Como 802.11b, la norma 802.11a se publicó en septiembre de 1999. Pero, en cambio, su capa Física puede trabajar a 5 GHz. La transmisión máxima es de 54 Mbps. Como en la anterior, las velocidades pueden ser a 48, 36, 24, 18, 12, 9 y 6 Mbps. A causa del cambio de frecuencia, las antenas 802.11a son incompatibles con las de 802.11b.

* 802.11g: Este estándar, ratificado en junio de 2003, es el sucesor del 802.11b. También

utiliza la banda de los 2,4 GHz y permite velocidades de 54 Mbps. Las velocidades posibles son las mismas que en 802.11a, es decir, 48, 36, 24, 18, 12, 9 y 6 Mbps.

* 802.11n: La especificación 802.11g sigue siendo, desde 2003, la más explotada comercialmente. IEEE ha finalizado la evolución 802.11n en septiembre de 2009. Después de las versiones 1.0 y 1.1, el grupo de trabajo de IEEE adoptó, en marzo de 2007, la versión 2.0 del borrador, que se acerca al estándar definitivo. En términos de capacidad, 802.11n en versión borrador incluía ya la calidad del servicio (QoS - Quality of Service), WMM (Wi-Fi MultiMedia) para las aplicaciones de VoIP (Voice over IP) y el streaming. Las evoluciones que se tenían que definir eran importantes, ya que se trataba de mejorar a la vez, considerablemente, la velocidad y la cobertura de radio. Se pusieron en marcha varios procedimientos y fue difícil garantizar sus definiciones en el ámbito del estándar. En primer lugar, se realizó un trabajo sobre la señal (capa Física). Este avance permitió prever una velocidad de 65 Mbps en lugar de los 54 Mbps de las especificaciones anteriores. La segunda mejora en las transmisiones la han realizado una serie de técnicas relativas a la tecnología MIMO (Multiple Input Multiple Output). Por multiplexado espacial, se pueden tratar simultáneamente hasta 4 flujos en lugar de uno solo. Utilizando más antenas de recepción que flujos, es posible recibir señales de varios caminos. La especificación 802.11n utiliza bandas de frecuencia de 2,4 y 5 GHz. Con esta última, es posible duplicar la longitud del canal, lo que permite ganar aún más velocidad. La velocidad máxima de la versión final de 802.11n es de 200 Mbps. Las técnicas utilizadas permiten en teoría alcanzar los 540 Mbps. El alcance en interiores es de 50 metros y de 125 metros en exteriores.

- *Hardware.*

La elección del hardware Wi-Fi requiere, en primer lugar, asegurarse de su compatibilidad con la norma de la capa Física.

* La tarjeta de red: Una adaptador Wi-Fi está compuesto por un chip conectado a una antena. Está integrado al equipo informático, ordenador portátil, PDA... o incluido en una tarjeta periférica. En el caso de una solución integrada, la antena también lo está. Por ejemplo, en un ordenador portátil se coloca a lo largo de la pantalla. Los fabricantes también pueden adoptar la tecnología Intel Centrino, que incluye Wi-Fi en una solución integral. Hay muchos modelos de tarjetas Wi-Fi que permiten incorporar esta función a un ordenador. Existen los formatos PC Card/PCMCIA, Compact Flash, PCI o USB.

* El equipo de infraestructura: Se utilizan principalmente dos dispositivos. Su objetivo es la interconexión de la red Wi-Fi a la red cableada Ethernet, lo que se denomina sistema de distribución (DS - Distribution System).

+ Punto de acceso (AP - Access Point): es el principal componente de la infraestructura de una red Wi-Fi. Como concentrador, centraliza todas las comunicaciones de los equipos que están asociados.

+ Puente (bridge) Wi-Fi: su función principal es interconectar dos redes cableadas Ethernet a través de la interfaz inalámbrica. Los puentes Wi-Fi ofrecen una solución de bajo presupuesto para conectar las redes Ethernet de diferentes edificios, sin tener que recurrir a la fibra óptica,

que, por supuesto, es más rápida.

- *Arquitectura.*

La primera arquitectura definida en la norma 802.11 permite una comunicación de igual a igual entre al menos dos equipos. Se denomina Independent Basic Service Set (IBSS), y se utiliza para crear redes ad hoc. La segunda arquitectura requiere un punto de acceso. En este caso Basic Service Set (BSS) actúa como administrador para las estaciones periféricas que se le asocian. Todas las comunicaciones deben pasar por él. Una red de mayor amplitud, con varios puntos de acceso, se llama Extended Service Set (ESS). Este tipo de red permite el desplazamiento por el interior de la empresa, asociándose sucesivamente a los puntos de acceso más cercanos y sin producir cortes de comunicación. Esta capacidad es la itinerancia o roaming.

- *Seguridad.*

* WPA: El Wi-Fi Protected Access se finalizó en 2003 en respuesta a las numerosas vulnerabilidades de WEP (Wired Equivalent Privacy). WPA implementa la mayor parte de las funcionalidades descritas en la norma 802.11i que se había completado totalmente en esa época. El diseño del protocolo se basa en la utilización de un servidor de autenticación basado en 802.1X que es necesario para la distribución de claves para cada uno de los usuarios. Existe una implementación más básica para particulares y pymes, que es la utilización de una clave compartida (Pre-Shared Key o PSK). WPA se basa normalmente en el protocolo TKIP (Temporal Key Integrity Protocol), que utiliza claves más largas que las claves WEP. Este protocolo TKIP permite el intercambio dinámico de las claves. Existen dos variantes de WPA (v1 o v2): WPA-personal. WPA-empresa. El WPA-personal no necesita servidor de autenticación; cada equipo se autentica directamente con el punto de acceso a través de una clave de 256 bits. La versión para empresas se basa en la utilización de un servicio RADIUS (Remote Dial In User Services) y se apoya necesariamente en un cifrado AES (Advanced Encryption Standard). De este modo, WPA garantiza no solamente la autenticación, sino también el cifrado. Igualmente protege la integridad de las tramas firmándolas. Así, es casi imposible realizar ataques de modificación de las tramas y de su CRC (Cyclic Redundancy Check) como con WEP. Ofrece igualmente un mecanismo para contar tramas que impide la repetición por parte de los atacantes. El algoritmo utilizado por la identificación de los mensajes es el MIC (Message Integrity Code), llamado también «Mickeal».

* WPA2: Se trata de la versión del protocolo que respeta escrupulosamente la norma IEEE 802.11i. El cifrado se implementa a través del protocolo AES. Desde 2006, es obligatoria la compatibilidad WPA2 para los equipos certificados Wi-Fi.

- *Utilización.*

En primer lugar se debe considerar esta tecnología como la versión inalámbrica de Ethernet. Se presenta como la respuesta a las exigencias de movilidad dentro de las empresas. Más allá del uso administrativo, que se traduce en un equipo colocado en una mesa, Wi-Fi permite una verdadera itinerancia. Otra utilización importante es la extensión de la red en la empresa. Llevar la red local allí donde aún no está disponible es, a partir de ahora, mucho más fácil que teniendo

que llevar el cable. Además de la facilidad y rapidez de la implementación, el coste representa un criterio de elección importante. Un Hot-Spot permite un acceso a Internet a través de la tecnología Wi-Fi. También se le conoce con el nombre de Acceso Público a Internet (API).

IEEE 802.15 y Bluetooth. Teniendo en cuenta a Dordogne (2015) el éxito de las tecnologías de red que utilizan la radio se confirma con Bluetooth, que se ha convertido en el estándar comercial de las redes de tipo RPAN (Radio Personal Area Network) y otras comunicaciones entre dispositivos.

- Antecedentes.

Lanzado en 1994 por la empresa Ericsson, esta tecnología lleva el nombre de un rey danés, Harald II. Apodado Harald II Blåtand (diente azul - blue tooth), fue un gran consumidor de arándanos que unificó Dinamarca, así como una parte de Suecia y Noruega, en el siglo IX. Este nombre señala la voluntad, por parte de Ericsson, de unificar el mundo de la telefonía móvil. Otros grandes fabricantes, como Nokia, Intel, Microsoft, Toshiba e IBM, se unieron a Ericsson a partir de 1998 para formar la Bluetooth Special Interest Group (SIG), que actualmente cuenta con más de 10.000 miembros. Destinado a permitir la comunicación entre equipos muy heterogéneos y poco distantes, Bluetooth integra, en un chip de menos de un centímetro, características muy avanzadas.

- Estandarización.

La especificación 1.0, publicada en julio de 1999, se adapta a las transmisiones de voz, datos e imágenes. Su velocidad teórica es de 1 Mbps. Su frecuencia de trabajo es la misma que la de Wi-Fi, 2,4 GHz; es muy económica en cuanto a energía. La segunda especificación data de finales del año 2004. Es todavía más económica en términos energéticos y, teóricamente, alcanza velocidades de 10 Mbps, lo que permite la transmisión de vídeos. El grupo IEEE 802.15 (WPANWG - Wireless Personal Area Network Working Group) eligió basarse en Bluetooth para su estándar. De hecho, se formaron 4 subdivisiones entre los grupos de proyecto o Tasks Groups. Del proyecto 802.15.1, publicado en junio de 2002, ha resultado un estándar basado en Bluetooth v1.1. El segundo, 802.15.2, que finalmente no ha salido a la luz, tenía como misión la coexistencia entre WPAN (802.15) y WLAN (802.11). La solución de alta velocidad (HR - High Rate) del grupo 802.15.3 se finalizó en junio de 2003. Permite la transferencia de archivos de vídeo y audio en streaming, basándose en la tecnología Ultra Wide Band (UWB), procedente del ejército estadounidense. Este estándar es capaz de alcanzar velocidades de varios cientos de megabits por segundo, a una distancia de varias decenas de metros. Además, esta solución permite atravesar obstáculos, como los muros de un edificio. Finalmente, en el otro extremo, la solución de baja velocidad, con un importante ahorro de energía, la del grupo 802.15.4, se aprobó en 2006. Esta norma, también llamado ZigBee, es la prolongación de HomeRF y puede utilizarse para transmitir comandos en lugar de datos. Su velocidad es de 250 Kbps como máximo a una distancia de unos 10 metros.

- Red Bluetooth.

Los equipos Bluetooth pueden interconectarse de dos maneras. La primera consiste en formar una única red, el piconet, que comprende un terminal administrador que asume hasta 7 terminales esclavos. Todas las comunicaciones, incluso entre esclavos, se administran y transitan por el administrador. Una segunda solución es interconectar varias redes en estrella. Estas forman una scatternet, en la cual el administrador de un piconet puede convertirse en el esclavo de otro piconet.

Bluetooth en realidad se diseñó para diferentes propósitos. Las funcionalidades de un dispositivo forman un repertorio de perfiles. Para que dos dispositivos se puedan comunicar, deben utilizar la misma funcionalidad. Entre los diferentes perfiles, podemos citar: GAP (Generic Access Profile), que define los procedimientos de búsqueda de dispositivos, de conexión y de seguridad. HS Profile (Headset Profile), para los kits de manos libres. LAN Access Profile. Fax Profile. FTP (File Transfer Profile). CTP (Cordless Telephony Profile). Una red TCP/IP de radio entre diferentes máquinas (Ad hoc) se puede construir gracias al perfil de red local de Bluetooth.

- *Clases de equipos.*

Existen tres clases de dispositivos Bluetooth:

- * Clase 1, de una potencia de 100 mW, para grandes distancias, de 100 metros.
- * Clase 2, de una potencia de 2,5 mW, para distancias medias, de entre 15 y 20 metros.
- * Clase 3, de una potencia de 1 mW, para pequeñas distancias, alrededor de 10 metros.

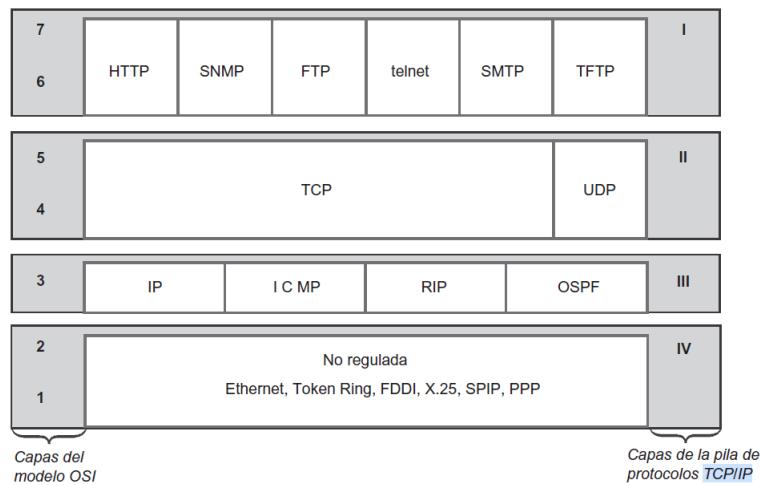
La mayor parte de los equipos comercializados son de clase 3. Esta tecnología se utiliza mucho para kits universales de manos libres para teléfonos móviles. También se encuentra en conexiones entre dispositivos (PDA...) o entre ordenadores y periféricos (impresoras...). En este último caso, sustituye a los cables USB.

Arquitectura de protocolos de red. - *Modelo de Referencia OSI:* según Olifer y Olifer (2009) la arquitectura de redes es una representación de la red como un sistema compuesto de varios elementos, cada uno de los cuales lleva a cabo una función específica. Todos los elementos de la red trabajan de manera coordinada con el fin de resolver la función común de interactuar entre las computadoras. En otras palabras, la arquitectura de red descompone un problema en una serie de sub-problemas que los elementos individuales de la red deben resolver. Uno de los elementos más importantes de la arquitectura de red es el protocolo de comunicaciones, el cual puede definirse como el conjunto formal de reglas para la interacción entre los nodos de la red. El desarrollo de la interconexión de sistemas abiertos (OSI, por sus siglas en inglés) fue un evento muy significativo en la estandarización de la arquitectura de las redes de computadoras. Dicho modelo, diseñado a principios de la década de 1980, resumió toda la experiencia acumulada de esa época. El modelo OSI representa un estándar internacional y define el método para descomponer verticalmente el problema de la interacción entre computadoras al delegar esta tarea a los protocolos de comunicaciones, los cuales se dividieron en siete capas (1. física, 2. enlace de datos, 3. red, 4. transporte, 5. sesión, 6.

presentación y 7. aplicación). Las capas de los protocolos de comunicaciones forman una jerarquía conocida como pilas de protocolos, en la que cada capa utiliza la capa inferior como una herramienta apropiada para resolver sus tareas. Las pilas de protocolos utilizadas en la actualidad (o las más populares hasta la fecha) reflejan en general la arquitectura del modelo OSI; sin embargo, cada pila de protocolos cuenta con características y diferencias específicas con respecto a la arquitectura del modelo OSI.

- *Modelo TCP/IP*: según Olifer y Olifer (2009) tiene una estructura multicapa; sin embargo, TCP/IP se diseñó antes que el modelo ISO/OSI. La correspondencia de las capas de TCP/IP respecto a las del modelo OSI es muy condicional. La pila de protocolos TCP/IP define cuatro capas: La capa de aplicación de TCP/IP (I) corresponde a las tres capas superiores del modelo OSI: las de aplicación, presentación y sesión; además, combina los servicios proporcionados por el sistema a las aplicaciones del usuario. Durante su operación en varias redes ubicadas en diferentes organizaciones y países, TCP/IP ha acumulado un gran número de protocolos y servicios de la capa de aplicación. La lista de dichos protocolos y servicios es muy larga e incluye protocolos ampliamente usados, como el protocolo de transferencia de archivos (FTP), el protocolo de emulación de terminal (telnet), el protocolo simple para la transferencia de correo (SMTP) y el protocolo de transferencia de hipertexto (http). Los protocolos de la capa de aplicación están instalados en los host. La capa de transporte de la pila de protocolos TCP/IP (II) puede proporcionar los dos tipos de servicios siguientes a la capa superior: entrega garantizada: protocolo de control de la transmisión (TCP) y entrega con el mejor esfuerzo: protocolo de datagrama del usuario (UDP). La capa de red, también llamada capa interred, se encuentra en el núcleo de la arquitectura TCP/IP (III). Las funciones de esta capa corresponden a las de la capa de red del modelo OSI y aseguran la transferencia de paquetes dentro de la interred, la cual se crea mediante la conexión de varias redes. Los protocolos de la capa de red soportan la interfase hacia la capa de transporte que está más arriba y reciben de ésta las solicitudes para la transmisión de datos utilizando la interred y de ahí hacia la capa de interfase de red. El protocolo Internet (IP) es el más importante de la capa de interred. Entre sus funciones se encuentra el envío de paquetes entre redes, de un ruteador a otro, hasta que el paquete llega a la red a la que está destinado. En contraste con los protocolos de las capas de aplicación y de transporte, IP está instalado no sólo en todos los host sino también en todas las puertas de enlace (Gateway). El IP es un protocolo de datagrama no orientado a la conexión que trabaja de acuerdo con el principio del mejor esfuerzo. Por último la capa de interfase de red de la pila de protocolos TCP/IP (IV) no está regulada estrictamente, es decir, soporta todas las tecnologías de red más comunes. Para las LAN, éstas son Ethernet, Token Ring, FDDI, Fast Ethernet y Ethernet Gigabit; para las WAN, éstas son protocolos punto a punto como el SLIP y el PPP; para las redes de commutación de circuitos, éstas son las tecnologías X.25, Frame Relay y ATM.

Tabla 10: Arquitectura de la pila de protocolos TCP/IP. Fuente: Olifer y Olifer (2009)



Internet. Según el Federal Networking Council (FNC) es un sistema de información global que está unido lógicamente en un único espacio de direccionamiento basado en el protocolo IP o en sus extensiones, permite el soporte de comunicaciones que emplean la familia de protocolos TCP/IP o sus extensiones, usa o proporciona acceso, de forma pública o privada, a servicios de alto nivel basados en las comunicaciones o estructuras antes mencionadas. Es la red de redes, o la WAN (Wide Area Network) más grande de la historia.

Direcciones IP. Según Kurose y Ross (2010) las direcciones IP tienen una longitud de 32 bits (lo que equivale a 4 bytes), por lo que existen un total de 2^{32} direcciones IP posibles. Aproximando 2^{10} a 10^3 , es fácil ver que hay unos 4.000 millones direcciones IP posibles. Estas direcciones normalmente se expresan utilizando la notación decimal con punto, en la que cada byte de la dirección se escribe en formato decimal y separada mediante un punto del resto de los bytes de la dirección. Por ejemplo, considere la dirección IP 193.32.216.9. El 193 es el número decimal equivalente a los 8 primeros bits de la dirección; el 32 es el equivalente decimal de los segundos 8 bits de la dirección, y así sucesivamente. Por tanto, la dirección 193.32.216.9 en notación binaria se expresa como sigue: 11000001 00100000 11011000 00001001.

Tabla 11: Clases de direcciones IP. Fuente: Olifer y Olifer (2009)

Clase	Primeros bits	Número de red más pequeño	Número de red más grande	Número de nodos
A	0	1.0.0.0 (0 – no usado)	126.0.0.0 (127 – reservado)	2^{24} (3 bytes)
B	10	128.0.0.0	191.255.0.0	2^{16} (2 bytes)
C	110	192.0.0.0	223.255.255.0	2^8 (1 byte)
D	1110	224.0.0.0	239.255.255.255	Direcciones multicast
E	11110	240.0.0.0	247.255.255.255	Reservado

Máscara de red. Según Olifer y Olifer (2009) para separar una dirección IP en un número de red y número de host, se utiliza la máscara de red asociada con esta dirección. La representación binaria de la máscara contiene unos en aquellos bits que deben interpretarse como el número de red en la dirección IP actual.

DHCP (Dinamyc Host Configuracion Protocol). Según Ossa y Ballesteros (2012) el protocolo de configuración dinámica es un estándar IP diseñado para simplificar la administración de la configuración de una red LAN, el estándar DHCP permite uso de servidores DHCP para administrar la asignación dinámica a los clientes DHCP de una red, de direcciones IP y otros detalles de configuración relacionados. Cada equipo de una red TCP/IP debe tener una dirección IP única. La dirección IP (junto con las máscara de subred relacionada) identifica al equipo host y a la subred a la que está conectado. El protocolo DHCP incluye tres métodos de asignación de direcciones IP:

- *Asignación manual o estática:* Asigna una dirección IP a una máquina determinada. Se suele utilizar cuando se quiere controlar la asignación de dirección IP a cada cliente, y evitar, también, que se conecten clientes no identificados.

- *Asignación automática:* Asigna una dirección IP de forma permanente a una máquina cliente la primera vez que hace la solicitud al servidor DHCP y hasta que el cliente la libera. Se suele utilizar cuando el número de clientes no varía demasiado.

- *Asignación dinámica:* el único método que permite la reutilización dinámica de las direcciones IP. El administrador de la red determina un rango de direcciones IP y cada dispositivo conectado a la red está configurado para solicitar su dirección IP al servidor cuando la tarjeta de interfaz de red se inicializa. El procedimiento usa un concepto muy simple en un intervalo de tiempo controlable. Esto facilita la instalación de nuevas máquinas clientes a la red.

SSH (Secure SHell). Según Ossa y Ballesteros (2012) es un protocolo que facilita las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor y que permite a los usuarios conectarse a un host remotamente. A diferencia de otros protocolos de comunicación remota tales como FTP o Telnet, SSH encripta la sesión de conexión, haciendo imposible que alguien pueda obtener contraseñas no encriptadas. Utiliza el puerto por defecto 22.

FTP (File Transfer Protocol). Según Ossa y Ballesteros (2012) el protocolo de transferencia de archivos es un protocolo de intercomunicación y transferencia de archivos entre equipos conectados a una red TCP, básicamente la estructura es cliente-servidor, este protocolo funciona en la capa de aplicación del modelo TCP/IP, los puertos de transferencia se manejan en el 20 y 21, un problema concurrente con este es que está pensado en máxima velocidad de transferencia, pero no a la máxima seguridad de la misma, ya que todo el proceso de login entre el cliente y el servidor se maneja en texto plano., con lo que un posible intruso en la red podría

estar capturando paquetes, coger nuestro user y pass, y muy posiblemente ingresar al server y apropiarse de nuestro archivos. Para solucionar el inconveniente se pueden utilizar herramientas como sftp o scp, que cifran la información, estas herramientas están disponibles en el paquete SSH

DNS (Domain Name Systems). Según Ossa y Ballesteros (2012) el sistema de nombres de dominio es un sistema de nomenclatura jerárquica para equipos de informática, servicio o cualquier recurso conectado a internet o una red LAN. La principal tarea de un servidor DNS es traducir tu nombre de dominio en una dirección IP , este servidor permite, una vez configurado, que tu web y tu correo electrónico sean localizados desde cualquier lugar del mundo mediante tu nombre de dominio. Es una base de datos distribuida, con información que se usa para traducir los nombres de dominio, fáciles de recordar y usar por las personas, en números de protocolo de Internet (IP) que es la forma en la que las máquinas pueden encontrarse en Internet.

HTTP (HyperText Transfer Protocol). Según Kurose y Ross (2010) el protocolo de transferencia de hipertexto es el protocolo de la capa de aplicación de la Web y se encuentra en el corazón de la Web. Está definido en los documentos [RFC1945] y [RFC 2616]. HTTP se implementa en dos programas: un programa cliente y un programa servidor. El programa cliente y el programa servidor, que se ejecutan en sistemas terminales diferentes, se comunican entre sí intercambiando mensajes HTTP. HTTP define la estructura de estos mensajes y cómo el cliente y el servidor intercambian los mensajes. Antes de explicar en detalle HTTP, vamos a hacer un breve repaso de la terminología Web. Una página web (también denominada documento web) consta de objetos. Un objeto es simplemente un archivo (como por ejemplo, un archivo HTML, una imagen JPEG, un applet Java o un clip de vídeo) que puede direccionarse mediante un único URL. La mayoría de las páginas web están constituidas por un archivo base HTML y varios objetos referenciados.

Según Tanenbaum y Wetherall (2012) la web (también conocida como WWW) comenzó en 1989 en el CERN, el Centro Europeo de Investigación Nuclear. En 1994, el CERN y el MIT firmaron un acuerdo para establecer el W3C (Consorcio World Wide Web, del inglés World Wide Web Consortium), una organización dedicada al desarrollo de web, a la estandarización de protocolos y a fomentar la interoperabilidad entre los sitios.

Según Olifer y Olifer (2009) la web es una estructura arquitectónica para tener acceso a documentos vinculados de manera distribuida sobre Internet. Así, desde el punto de vista del usuario, la WWW consiste en un enorme conjunto de documentos a nivel mundial que se conoce como páginas web, las cuales pueden tener vínculos a otras páginas web; a esto se le conoce como hipertexto. Las páginas web pueden ser visualizadas mediante un “navegador”, como por ejemplo el Internet Explorer.

Servicio hipermedia: WWW (World Wide Web), teniendo en cuenta a Dordogne (2015) HTTP es, sobre todo, un protocolo de transferencia de archivos, se utiliza para formatear y

visualizar. Los archivos transmitidos al cliente los interpreta un software navegador (browser). El protocolo HTTP es utilizado por un servidor Web, que almacena la información en forma de páginas de texto (HTML), imágenes, vídeos, sonidos, etc. Cada entidad corresponde a un archivo, dentro de una jerarquía. La versión de HTML que actualmente está es la versión 5, data del año 2014. Sin embargo, la mayor parte de los navegadores de Internet, en sus últimas versiones, incorporan las novedades ofrecidas por esta versión del protocolo (Edge, Chrome, Firefox, Safari u Opera). La URL <http://html5test.com/> permite probar las funcionalidades HTML5 que soporta su navegador. HTML5 proviene de la colaboración entre el W3C (World Wide Web Consortium), que sobre todo ha trabajado en XHTML 2.0, y de WHATWG (Web Hypertext Application Technology Group), que se centró en los formularios y aplicaciones web. Se han enunciado algunas nuevas reglas para esta nueva versión: Las nuevas funcionalidades deben basarse en HTML, CSS (Cascading Style Sheetu hojas de estilo en cascada), DOM (Document Object Model) y JavaScript. Se debe reducir al máximo la utilización de componentes externos (p. ej., Plugin Flash). Debe haber un perfecto control de errores. Numerosas etiquetas complementarias reemplazan a scripts.

```
<!DOCTYPE html>
<html>
<head> <title>Título del documento</title> </head>
<body> El contenido del documento...</body>
</html>
```

HTML5 debe ser independiente de los dispositivos. Los procesos de desarrollo deben ser accesibles al público. En HTML5 solo hay una declaración <!DOCTYPE>. De este modo, el documento HTML5 más pequeño posible se parecerá a este: Entre las nuevas funcionalidades de HTML5, se encuentran:

- El elemento <canvas> para el diseño 2D.
- Los elementos <audio> y <video> para las funcionalidades multimedia.
- La implementación del almacenamiento local.
- Nuevos elementos que definen nuevos contenidos, como <article>, <section>, <nav> o <footer>.
 - La etiqueta <nav> permite definir vínculos de navegación.
 - La etiqueta <footer> es la firma o el pie de página de un documento.
 - Nuevos formularios, como calendar, date, time, email, search o url.

Para visualizar esta información, se utilizan las URL (Uniform Resource Locator). Una URL requiere, en primer lugar, el protocolo (<http://>) y a continuación el alias del servidor Web y la referencia de la entidad.

Por ejemplo, un navegador puede mostrar la página «redes.html» del directorio «Libros» llamando a la URL <http://www.editions-eni.fr/libros/redes.html>, que está gestionada por el servidor www.editions-eni.fr.

Para formular las comunicaciones entre clientes y servidores, el protocolo HTTP utiliza

comandos, llamados métodos, en el puerto TCP 80. Estos métodos ofrecen diferentes funciones, como la llamada a páginas (get), el envío de formularios (post). El protocolo que se utiliza actualmente es HTTP 1.1, de forma predeterminada, el texto transmitido entre un cliente y un servidor se realiza sin cifrar. De esta manera es perfectamente legible por cualquiera que intercepte la conversación. Para solucionar este defecto, la versión segura del protocolo, HTTPS (TCP 443), cifra la comunicación y convierte la información en confidencial. El protocolo HTTPS también se denomina SSL (Secure Socket Layer). Su versión estándar, que se puede usar con otros protocolos, es TLS (Transport Layer Security). Esta denominación muestra una voluntad más general de cifrar la comunicación.

SMTP (Simple Mail Transfer Protocol). Según Ossa y Ballesteros (2012) el protocolo simple de transferencia de correo es un protocolo estándar que permite la transferencia de correo de un servidor a otro, éste es un protocolo que funciona en línea y es encapsulado en una trama TCP/IP. El correo se envía directamente al servidor de correo del destinatario, este protocolo funciona con comandos de texto enviados al servidor SMTP. A cada comando enviado por el cliente (validado por una cadena de caracteres ASCII que equivalen a presionar la tecla Enter) le sigue una respuesta del servidor SMTP compuesta por un número y un mensaje descriptivo.

POP3 (Post Office Protocol). Según Ossa y Ballesteros (2012) el protocolo de oficina de correos es como su nombre lo indica, permite recoger el correo electrónico en un servidor remoto (servidor POP). Es utilizado por personas que no están permanentemente conectadas a Internet, ya que así pueden consultar sus correos electrónicos recibidos sin que estén conectados. POP3 está diseñado para recibir correo, no para enviarlo; les permite a los usuarios con conexiones intermitentes o muy lentas, descargar su correo electrónico mientras tienen conexión y revisarlo posteriormente incluso estando desconectados.

IMAP (Internet Message Access Protocol). Según Ossa y Ballesteros (2012) es un protocolo de red de acceso a mensajes electrónicos almacenados en un servidor. Mediante IMAP se puede tener acceso al correo electrónico desde cualquier equipo que tenga una conexión a Internet. El protocolo IMAP fue diseñado para solucionar las carencias del protocolo POP, principalmente la movilidad y el procesamiento de correo desde diferentes puestos.

Transmisión de vídeo en tiempo real (video streaming).

Definiciones.

Video. Teniendo en cuenta of Art (2012) el video, o imagen en movimiento en general, se crea a partir de una secuencia de imágenes fijas llamadas cuadros o fotogramas. Al grabar y luego reproducir los cuadros en rápida sucesión, se crea una ilusión de movimiento. El video se

puede editar eliminando algunos cuadros y combinando secuencias de cuadros, llamados clips, juntos en una línea de tiempo.

Formato de video (video format). Según of Art (2012) un formato de video define la forma en que se graba y almacena el video. Normalmente se especifica: Codec/compresor, cuadros por segundo, tamaño del cuadro, relación de aspecto del cuadro, relación de aspecto de píxeles, método de escaneo (entrelazado o progresivo). Los formatos comunes son DV, HDV y AVCHD. Los formatos basados en cinta, como DV y HDV, pueden transferirse a una computadora para editarlos a través de Firewire. Los formatos basados en archivos, como AVCHD, ya se almacenan como archivos y se pueden transferir a una computadora para editarlos a través de USB. Es posible que los formatos basados en archivos deban convertirse durante o después de la transferencia para que sean compatibles con el software de edición.

Códec (compresor). Según of Art (2012) es la abreviatura de codificador-decodificador y describe el método en el que los datos de video se codifican en un archivo y se decodifican cuando se reproduce el archivo. La mayoría de los videos se comprimen durante la codificación, por lo que los términos códec y compresor se usan indistintamente. La transcodificación es el proceso de conversión de un códec a otro. Los códecs pueden ser sin pérdida, lo que significa que no desechan ningún dato, o Pérdida, lo que significa que los datos se pierden durante la codificación. Los códecs sin pérdida son de mayor calidad que los códecs con pérdida, pero producen archivos de mayor tamaño. En un flujo de trabajo de video, debe evitar la transcodificación a un códec con pérdida hasta la salida final. Esto significa que su edición final se puede exportar a H.264 para la web, pero no debe usar el archivo H.264 para una edición posterior, creación de DVD, etc.

Cuadro (frame). De acuerdo a of Art (2012) es una sola imagen fija dentro de un videoclip. Al igual que con cualquier imagen digital, un cuadro consta de píxeles (elementos de imagen), y cada píxel representa un color dentro de la imagen. Cuanto mayor sea el número de píxeles, más exactamente se puede representar una imagen. Esto se llama resolución y se mide en megapíxeles.

Tamaño del cuadro (frame size). Según of Art (2012) describe el tamaño de un solo cuadro de video: ancho x alto, medido en píxeles. El ancho del marco puede variar dependiendo de si los píxeles en el marco son píxeles cuadrados o no cuadrados. Un ejemplo: un cuadro DV-PAL 4:3 tiene 768 píxeles de ancho (píxeles cuadrados) o 720 píxeles de ancho (píxeles no cuadrados). Esto se debe a que se requiere un número menor de píxeles para crear el mismo cuadro de tamaño en la pantalla si cada píxel individual es más ancho. Los gráficos de computadora usan píxeles cuadrados, mientras que las cámaras de video usan píxeles no cuadrados. Por lo tanto, si crea gráficos para video, use el tamaño de marco de píxeles cuadrados. Una relación de aspecto de

píxeles cuadrados se puede escribir como 1.0 (ancho ÷ altura). Todos los demás valores indican píxeles no cuadrados.

Relación de aspecto del cuadro (frame aspect ratio). De acuerdo a of Art (2012) la relación de aspecto del cuadro describe la relación entre el ancho y el alto de un solo cuadro de video. El video es horizontal, por lo que el ancho de un cuadro es mayor que la altura. Las relaciones de aspecto de fotogramas típicas para video son 4:3 y 16:9. Las cámaras fotográficas digitales a menudo usan 4:3 o 3:2. La relación 4:3 se refiere como estándar. La relación 16:9 se conoce como pantalla panorámica, y en ocasiones se llama (de manera inexacta) anamórfico . Algunos videos de pantalla ancha son anamórficos, pero no todos.

Cuadros por segundo (frame rate). Según of Art (2012) la velocidad de fotogramas le indica cuántos fotogramas por segundo hay cuando graba o reproduce un video. Las cámaras de video en Europa usan 25 cuadros por segundo (fps). En USA y Japón se utilizan 29.97fps o 30fps. La animación funciona al grabar cada fotograma individualmente (por ejemplo, con una cámara fotográfica fija) y luego reproducirlos a una velocidad de fotogramas. Los animadores a menudo trabajan con una velocidad de fotogramas más baja (por ejemplo, 12 fps), por lo que se necesitan menos fotogramas para el mismo clip de video. Si cambia la velocidad de fotogramas de un videoclip de 12 fps a 25 fps, por ejemplo. Al agregarlo a un proyecto de edición de 25 fps, cada fotograma se repetirá para mantener el clip la misma duración.

Método de exploración (scanning method). Según of Art (2012) se muestra un cuadro de video en una pantalla al escanear cada línea horizontal de izquierda a derecha para formar la imagen. El método de escaneo describe la forma en que se hace esto, ya sea escaneando primero las líneas impares, luego las líneas pares (entrelazadas), o escaneando cada línea en orden (progresivo).

- *Exploración entrelazada (scan interlaced):* Según of Art (2012) la exploración entrelazada describe un método de grabación de video en el que cada cuadro consta de dos campos, que se combinan (entrelazan) para formar un cuadro completo (denominado i. Ejemplo: 1080i). Los televisores SD y los reproductores de DVD utilizan el escaneo entrelazado para mostrar el video. El campo superior (campo 1) contiene las líneas impares que forman la imagen. El campo inferior (campo 2) contiene las líneas pares que forman la imagen.

- *Exploración progresiva (scan progressive):* Según of Art (2012) la exploración progresiva o progresiva describe un método de grabación de video donde cada cuadro se graba como una imagen completa, sin campos (denominado p. Ejemplo: 1080p). Esto es similar a cómo una película o una cámara fotográfica graba imágenes, por lo que a menudo se dice que el video progresivo se ve más como una película. Progresivo es a menudo una opción de grabación en cámaras de video HD. Es una buena opción para grabar video que se ralentizará, ya que evita que aparezcan artefactos de entrelazado en la imagen ralentizada. Los monitores de computadora

y algunos televisores de alta definición utilizan el escaneo progresivo para mostrar el video. Todos los videos para reproducir en una computadora (por ejemplo, en la web) deben ser de exploración progresiva.

Arquitectura de protocolos de un sistema de video streaming.

RSTP. Según Arlandiz (2009) el protocolo de streaming de tiempo real (RTSP Real Time Streaming Protocol) es un estándar desarrollado por la Universidad de Columbia en el RFC 2326 para el IETF. El RTSP es un protocolo cliente-servidor de nivel de aplicación para el control del flujo de datos en tiempo real. El RTSP es básicamente un protocolo de señalización o de control, pero también provee de algunas funcionalidades de calidad de servicio. El RTSP utiliza el RTP como protocolo subyacente para la entrega de datos y ofrece al usuario los controles VCR: Play, Stop, Pause, FF, REW; así como, acceso aleatorio a cualquier corte del media. El RTSP facilita al servidor el ajuste de la velocidad de transmisión de los datos al ancho de banda de la conexión ante congestiones en la red, con el fin de utilizar lo mejor posible la capacidad disponible. Una vez, establecido, puede controlar uno o varios streams sincronizados temporalmente de los diferentes medias, como el audio y el video. Las fuentes de datos pueden incluir tanto datos fijos como datos en tiempo real. El RTSP proporciona mecanismos para seleccionar canales de envío (como el UDP, UDP multic平, TCP) y mecanismos de entrega continua de streams basados en RTP. Se puede considerar que el RTSP actúa “como control remoto de la red” para los servidores multimedia.

RTP. Según Arlandiz (2009) es el protocolo más importante normalizado para Streaming. Todos los streams de contenidos media, sin tener en cuenta su formato y contenido, son encapsulados en paquetes RTP. El RTP dispone de varios campos para datos que no se encuentran presentes en TCP, en particular un Timestamp y un número de Secuencia. RTP funciona sobre UDP y utiliza sus aplicaciones de multiplexado y checksums, aunque puede soportar otros protocolos de transporte. Permite el control del servidor media para que los streams de video sean servidos a la velocidad adecuada. El reproductor media está, entonces, en condiciones de poder organizar los paquetes RTP recibidos en el orden correcto y reproducirlos a la velocidad adecuada. El RTP transmite paquetes en tiempo real tanto de audio como de video. RTP no garantiza por sí sólo, la entrega de los datos en tiempo real, pero provee de los mecanismos de envío y recepción necesarios a las aplicaciones, para que puedan soportar los datos streaming. El RTP ofrece entrega de datos multicast siempre que la red subyacente ofrezca servicios de multicast.

RTCP. Según Arlandiz (2009) es usado conjuntamente con el RTP y utiliza TCP para la conexión bidireccional cliente-servidor. Es un protocolo que da calidad de servicio al sistema y proporciona mecanismos de realimentación para informar de la calidad en la distribución de

los datos. Eso, implica la transmisión periódica de paquetes de control a todos los participantes de una sesión. Los mensajes, que se reciben, incluyen informes relativos al número de paquetes perdidos y estadísticas de «jitter». Esta información puede ser utilizada por aplicaciones de nivel superior para el control de la sesión y mejora de la transmisión, por ejemplo, el bit/rate de un stream puede modificarse para combatir la congestión de red, se puede diagnosticar fallos en la distribución o se puede usar para controlar un mecanismo adaptativo de codificación, que responda a las condiciones de la red. Los paquetes de RTCP se envían de modo que el tráfico en la red no aumenta linealmente con el número de agentes participantes en la sesión, es decir el intervalo de envío se ajusta de acuerdo al tráfico.

HTTP. El HTTP según Arlandiz (2009) es el protocolo del mundo Web (WWW), usado en cada transacción. HTTP significa Hyper Text Transfer Protocol, es decir, protocolo de transferencia de hipertextos a nivel de aplicación. El hipertexto es el contenido de las páginas web, y el protocolo de transferencia es el sistema mediante el cual se envían las peticiones para acceder a una página web, y las respuestas del servidor de esa web, remitiendo al usuario la información que se visualizará en pantalla. También, sirve el protocolo para enviar información adicional en ambos sentidos, como formularios con mensajes y otros similares. El HTTP es un protocolo del tipo stateless, es decir, que no guarda ninguna información sobre conexiones anteriores. Al finalizar la transacción, todos los datos se pierden. Por esto se popularizaron las cookies, que son pequeños ficheros guardados en el propio ordenador que puede leer un sitio web al establecer conexión con él, y de esta forma reconocer a un visitante que ya estuvo en ese sitio anteriormente. Gracias a esta identificación, el sitio web puede almacenar gran número de información sobre cada visitante, ofreciéndole así un mejor servicio. A parte de las cookies, se están desarrollando nuevas tecnologías como ActiveX, Java, JavaScript para poder hacer de manera más inteligente las interacciones con el usuario. Desde el punto de vista de las comunicaciones, HTTP está soportado sobre los servicios de conexión TCP/IP, y funciona de la misma forma que el resto de los servicios comunes de los entornos UNIX: un proceso servidor escucha en un puerto de comunicaciones TCP (por defecto, el 80), y espera las solicitudes de conexión de los clientes Web. Una vez que se establece la conexión, el protocolo TCP se encarga de mantener la comunicación y garantizar un intercambio de datos libre de errores. Además, los paquetes HTTP están protegidos con checksums. De esta manera, los paquetes dañados o perdidos son nuevamente retransmitidos, por lo que todos los archivos recibidos son totalmente restaurados. HTTP puede utilizarse también para la descarga de contenidos media, especialmente si los archivos son de pequeño tamaño y el número de usuarios concurrentes es limitado. Si la velocidad de la conexión es inferior a la necesaria para la transferencia del media, éste todavía puede ser reproducido aunque no uniformemente. El tiempo de transferencia del archivo descargado depende de su tamaño y de la velocidad de la conexión. La primera versión de HTTP (HTTP/0.9) fue un protocolo simplificado para la transferencia de datos de baja calidad a través de Internet. La versión

HTTP/1.0 mejoró el protocolo permitiendo mensajes en formato MIME (Extensiones 73/161 Multipropósito de Correo Internet), metadatos sobre los datos transferidos y modificadores ante la semántica petición/respuesta. De esta forma, el protocolo puede intercambiar cualquier tipo de dato, sin preocuparse de su contenido. La transferencia se realiza en modo binario, byte a byte, y la identificación MIME permitirá que el receptor trate adecuadamente los datos conteniendo. La especificación HTTP/1.1 establecía funciones relacionadas con Proxy, caching, conexiones mantenidas y host virtuales. Actualmente, HTTP dispone de una variante cifrada en SSL (Secure Sockets Layer), llamada HTTPS. HTTP/1.1 incorpora requerimientos más rigurosos que HTTP/1.0, con el fin de asegurar la implementación fiable de sus prestaciones. Ello, permite más funcionalidades que la simple recuperación, incluyendo búsqueda, refresco y comentarios. HTTP facilita un conjunto de métodos sin limitaciones que indica el propósito de una petición. Se construye sobre la disciplina de referencia suministrada por la Uniform Resource Identifier (URI), como una localización (URL) o un nombre (URN), indicando el recurso sobre el que es aplicado. Los mensajes son enviados en un formato similar al utilizado por el correo de Internet, según lo define el MIME. HTTP se utiliza como un protocolo genérico para comunicaciones entre gestores de usuarios y servidores proxy/pasarelas a otros sistemas en Internet, incluidos los soportados por SMPT, NNTP, FTP y otros protocolos. De esta forma, HTTP facilita acceso básico de hipermedia (hipermedia-hipertexto) a otros recursos disponibles desde diversas aplicaciones.

Herramientas de transmisión de video sobre HTTP.

Motion. Según GNU/Linux (2019) es un programa para el S.O. GNU/Linux altamente configurable que monitorea las señales de video de muchos tipos de cámaras, se distribuye principalmente bajo la GNU General Public Licence (GPL) versión 2 o posterior. Utilizado para monitorear cámaras de seguridad, observar pájaros, vigilar mascotas, crear videos de timelapse y más.

- Motion permite: crear videos o guardar fotos de la actividad, grabar de paso de muchas cámaras IP, ver transmisión en vivo de cámaras mediante el protocolo http, invocar guiones cuando ocurren actividades, registrar actividad en múltiples tipos de bases de datos, totalmente personalizables para privacidad o detección de movimiento, soporte completo de tls (https) con autenticación para control web y transmisiones.

- Soporta muchos tipos de dispositivos: cámaras de red a través de RTSP, RTMP y HTTP, cámaras PI, cámaras web V4L2, tarjetas de captura de video y archivos de películas existentes

Mjpeg-streamer. De acuerdo a Stoveken (2018) es una aplicación de línea de comandos que copia fotogramas JPEG de uno o más complementos de entrada a múltiples complementos de salida para el S.O. GNU/Linux. Se puede usar para transmitir archivos JPEG a través de una red basada en IP desde una cámara web a varios tipos de espectadores como Chrome,

Firefox, Cambozola, VLC, mplayer y otros programas capaces de recibir transmisiones MJPG. Originalmente fue escrito para dispositivos integrados con recursos muy limitados en términos de RAM y CPU. Su antecesor "uvc_streamer" se creó porque las cámaras compatibles con Linux-UVC producen directamente datos JPEG, lo que permite la transmisión rápida y perfecta de M-JPEG incluso desde un dispositivo integrado que ejecuta OpenWRT. El módulo de entrada "input_uvc.so" captura dichos cuadros JPG de una cámara web conectada. Mjpg-streamer ahora admite una variedad de dispositivos de entrada diferentes. Al iniciar mjpg-streamer, especifica uno o más complementos de entrada y un complemento de salida. Por ejemplo, para transmitir una cámara web compatible con V4L a través de un servidor HTTP (el caso de uso más común), puede hacer algo como esto: mjpg_streamer -i input_uvc.so -o output_http.so y cada complemento admite varias opciones, puede ver las opciones del complemento a través de su opción –help: mjpg_streamer -i 'input_uvc.so –help'

Cámara web.

Definición. Teniendo en cuenta a EcuRed (2019) es una cámara digital conectada a una computadora, la cual puede capturar imágenes y transmitirlas a través de Internet, ya sea a una página Web o a otra u otras computadoras de forma privada (periférico de entrada). Las cámaras Web necesitan una computadora para transmitir las imágenes. Sin embargo, existen otras cámaras autónomas que tan sólo necesitan un punto de acceso a la red informática, bien sea ethernet o inalámbrico. Para diferenciarlas las cámaras Web se las denomina cámaras de red.

Características. De acuerdo a EcuRed (2019):

- Tiene una resolución por lo general baja, aproximadamente 640x480 píxeles, ya que las imágenes transmitidas instantáneamente por Internet deben de tener un tamaño muy bajo archivo.
- Dependiendo el modelo, tienen la lente giratoria de hasta 360° horizontales, una base adaptable a la superficie, e incluso micrófono integrado.
- Pueden tomar fotos al instante pero con baja resolución.
- Su diseño es muy específico para aplicaciones de entretenimiento y en algunos casos como cámara de vigilancia.

Software. Según EcuRed (2019) la instalación básica de una cámara web consiste en una cámara digital conectada a una computadora, normalmente a través del puerto USB. Lo que hay que tener en cuenta es que dicha cámara no tiene nada de especial, es como el resto de cámaras digitales, y que lo que realmente le da el nombre de "cámara web" es el software que la acompaña. El software de la cámara web toma un fotograma de la cámara cada cierto tiempo (puede ser una imagen estática cada medio segundo) y la envía a otro punto para ser visualizada.

Tecnología. Teniendo en cuenta a EcuRed (2019) las cámaras web normalmente están formadas por una lente, un sensor de imagen y la circuitería necesaria para manejarlos. Existen distintos tipos de lentes, siendo las lentes plásticas las más comunes. Los sensores de imagen pueden ser CCD (charge coupled device) o CMOS (complementary metal oxide semiconductor). Este último suele ser el habitual en cámaras de bajo coste, aunque eso no signifique necesariamente que cualquier cámara CCD sea mejor que cualquiera CMOS.

Las cámaras web para usuarios medios suelen ofrecer una resolución VGA (640x480) con una tasa de unos 30 fotogramas por segundo, si bien en la actualidad están ofreciendo resoluciones medias de 1 a 1,3 MP, actualmente las cámaras de gama alta cuentan con 3, 5, 8, 10 y hasta 15 megapixeles y son de alta definición.

La circuitería electrónica es la encargada de leer la imagen del sensor y transmitirla a la computadora. Algunas cámaras usan un sensor CMOS integrado con la circuitería en un único chip de silicio para ahorrar espacio y costes. El modo en que funciona el sensor es equivalente al de una cámara digital normal. También pueden captar sonido, con una calidad mucho menor a la normal.

Sistemas embebidos o empotrables.

Definición. Teniendo en cuenta a Arriarán (2015) el término “sistema embebido” hace referencia a todo circuito electrónico digital capaz de realizar operaciones de computación, generalmente en tiempo real, que sirven para cumplir una tarea específicas en un producto. Los sistemas embebidos no son equivalentes a los sistemas de cómputo usados en las laptops o en computadoras de escritorio que se venden en las tiendas tecnológicas, ya que los sistemas embebidos suelen tener recursos limitados y aplicaciones específica que los hacen sumamente útiles en múltiples ambientes.

Teniendo en cuenta a González (2018) sistema empotrado o embebido es un sistema de computación de tiempo real acrítico o “blando” diseñado a medida para resolver unas necesidades específicas. Un sistema empotrado está compuesto, de forma integrada por lo general, por varios de los siguientes componentes: CPU (microprocesador, microcontrolador, DSP, etc), fuente de alimentación, baterías, memoria, salidas y entradas analógicas y digitales, conversores dc/dc, filtros, reloj en tiempo real, etc. Sistemas complejos pueden funcionar con sistemas operativos.

Función. De acuerdo a González (2018) las principales funciones de un sistema empotrado son la adquisición de datos, tratamiento y visualización, analítica, conectividad avanzada, supervisión de la condición remota, mantenimiento predictivo, etc.

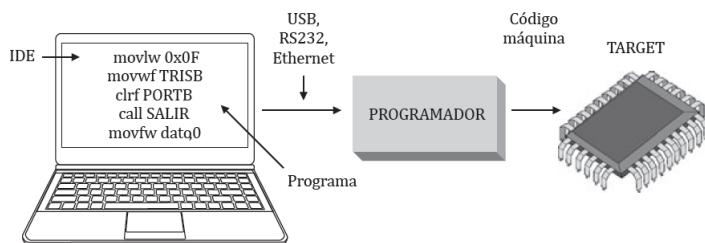
Arquitectura. De acuerdo a Arriarán (2015) la arquitectura de un sistema embebido contiene un microprocesador dedicado capaz de ejecutar instrucciones a una determinada

velocidad, controlada por una señal de reloj. De acuerdo con la arquitectura del microprocesador del sistema embebido, los recursos internos (periféricos) y la máxima frecuencia de operación, se define la potencia de procesamiento, este parámetro se mide en unidades de MIPS (millones de instrucciones por segundo).

Lenguaje de programación. Según Arriarán (2015) el programa que ejecuta un sistema embebido es por lo general elaborado en lenguajes como el ensamblador, ANSI C, C++ o Basic. Algunos sistemas embebidos tienen la capacidad de ejecutar sistemas operativos limitados, conocidos como RTOS (Real Time Operating Systems), que permiten que el procesador ejecute diversas tareas, asignándoles una prioridad y un orden de ejecución.

Teniendo en cuenta a Arriarán (2015) el proceso de programación de un sistema embebido requiere inicialmente de un computador o Host, en el cual se ejecute una herramienta de software denominada IDE (Integrated Development Environment). Dentro de esta herramienta, se realiza la programación de instrucciones del sistema embebido, que, por lo general, puede incluir varios archivos fuente con partes del programa. La secuencia de instrucciones que especifica el programa de un archivo de texto que será grabado en el sistema embebido se conoce como firmware. Cuando el programa es finalizado, se hace uso de un software llamado “compilador”. Esta herramienta transforma el texto que contiene las instrucciones de forma secuencial y lo convierte en un código de máquina: un archivo compuesto de números binarios que va a ser enviado directamente a la memoria de programa del sistema embebido (denominado Target). Usualmente, se requiere una interfaz entre el Host y el Target, conocida como “el programador”, esta interfaz es un equipo que se puede conectar al Host a través del puerto RS232, USB, Ethernet o WiFi.

Figura 23: Interfaz Host-Target para programación del sistema embebido. Fuente: Arriarán (2015)



El proceso de desarrollo. De acuerdo a Guillén (2011) la concepción del sistema en su totalidad es parte de una idea de una persona, o de un equipo de diseño o de un encargo a medida hecho para un "cliente" que quiere resolver un problema determinado.

Podemos establecer un serie de tareas previas que culminarán en la elaboración de un anteproyecto, sobre la base del cual se tomarán las decisiones de seguir adelante o descartar el proceso:

- 1) Determinación de los requisitos globales del sistema.

- 2) Selección del microprocesador, microcontrolador o DSP más adecuados.
- 3) Selección de la tecnología de fabricación más adecuada.
- 4) Elección de la memoria y del sistema operativo, si procede.
- 5) Determinación de las entradas/salidas, comunicaciones, etc.
- 6) Determinación de las necesidades de homologación en función de la aplicación.
- 7) Selección del equipo humano más oportuno para su desarrollo, selección de proveedores hardware y software, etc.
- 8) Elaboración de un anteproyecto, cuanto más detallado mejor, que nos permita evaluar la viabilidad técnica y económica del sistema, y también generar un presupuesto de costes tan veraz como sea posible.

Fases de diseño. a) Diseño inicial del sistema, que incluye toda una serie de tareas que acabarán en la elaboración de un esquema eléctrico del sistema y en un diseño de necesidades de software.

b) A partir del esquema y de la forma física de cada uno de los componentes que intervienen en él, diseño hardware del sistema. Esta tarea incluye el posicionamiento de cada uno de los componentes y el encaminamiento de las pistas conductoras que llevarán a cabo las interconexiones necesarias entre los pivotes de los componentes, lo que generará un prototipo de placa de circuito impreso (PCB). A partir de esta PCB, una vez producida, se lleva a cabo el montaje o el ensamblaje de todos y cada uno de los dispositivos mediante el procedimiento de soldadura más adecuado. Este proceso acaba en un prototipo de hardware.

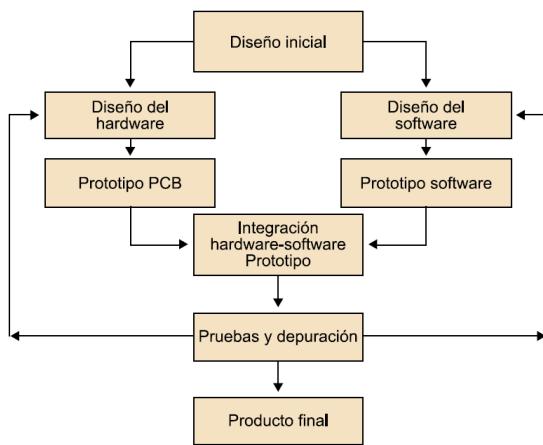
c) Desarrollo del prototipo de software con la programación inicial del microcontrolador o de los microcontroladores que formen parte del sistema empotrado.

d) Integración del hardware/software mediante el vertido o la programación en el circuito de los microcontroladores. Así, se dispondrá del primer prototipo preparado para proceder a su test y depuración.

e) Pruebas y depuración del software y hardware mediante el uso de prototipos hasta llegar a la versión final. Si se detectan errores en el hardware, será necesario rediseñar la placa y volver a empezar el proceso. Si los errores son de software, el proceso es similar, salvo que es menos costoso en cuanto a materiales que en cuanto a horas de ingeniería.

f) Obtención del producto final. Después del resultado satisfactorio en todas las pruebas se conseguirá el producto final.

Figura 24: Diagrama de fases para el diseño de un sistema empotrado. Fuente: Guillén (2011)



Placas bases, módulos, periféricos y fuentes de alimentación. Las partes electrónicas están presentadas en consonancia a las necesidades del dispositivo propuesto.

Placas bases embebidas para la unidad de control. En el mercado actual existen una variedad amplia de placas bases embebidas, en este punto se consideraron y estudiaron los fabricantes de: Arduino, Raspberry Pi y Beagleboard.org, por ser placas ya consolidadas, bien documentadas y con una comunidad extendida para proyectos de desarrollo de sistemas embebidos.

Arduino (placa de desarrollo). De acuerdo a Arduino (2019) es una plataforma electrónica de código abierto basada en hardware y software, fácil de usar. Las placas Arduino pueden leer entradas (luz en un sensor, un dedo en un botón o un mensaje de Twitter) y convertirlo en una salida: activar un motor, encender un LED, publicar algo en línea. Puede decirle a su placa qué hacer enviando un conjunto de instrucciones al microcontrolador de la placa. Para hacerlo, utiliza el lenguaje de programación Arduino (basado Wiring) y el software Arduino (IDE), basado en Processing.

Con los años, Arduino ha sido el cerebro de miles de proyectos, desde objetos cotidianos hasta instrumentos científicos complejos. Una comunidad mundial de creadores (estudiantes, aficionados, artistas, programadores y profesionales) se ha reunido en torno a esta plataforma de código abierto, sus contribuciones se han sumado a una increíble cantidad de conocimiento accesible que puede ser de gran ayuda para principiantes y expertos por igual.

Nació como un proyecto para estudiantes en el instituto de diseño interactivo IVREA (Ivrea, Italia) en 2005, con el objetivo de introducir a la electrónica a aquellas personas que no tenían conocimientos previos en ciencias. Allí fue donde el estudiante colombiano Hernando Barragán, basándose en la herramienta de programación de microcontroladores PIC's diseñada por el docente Massimo Banzi, desarrolló la primera tarjeta electrónica Wiring, el lenguaje de programación y el entorno de desarrollo. Poco tiempo después, varios investigadores, entre los

que se encontraba el español David Cuartielles, aunaron sus esfuerzos para hacer esa tarjeta más económica, ligera y accesible, creando la tarjeta Arduino.

En cuanto al software es de código abierto, su mayor virtud es su sencillez. Consiste en un entorno de desarrollo integrado de código abierto que implementa el lenguaje de programación Processing/Wiring. Este lenguaje de fácil utilización está basado en Java y el entorno de desarrollo se puede descargar gratuitamente. Además, se dispone de numerosas librerías que abstraen y facilitan la tarea de la programación.

Existen una cantidad de placas Arduinos como: Arduino Galileo, Arduino Uno, Arduino Leonardo, Arduino Due, Arduino Tre, Arduino Zero, Arduino Micro, Arduino Esplora, Arduino Mega ADK, Arduino Ethernet, Arduino Mega 2560, Arduino Robot, Arduino Mini, Arduino Nano LilyPad, Arduino Simple LilyPad, Arduino SimpleSnap LilyPad, Arduino LilyPad, Arduino USB, Arduino Pro Mini, Arduino Fio, Arduino Pro, Arduino MKR1000/Genuino MKR1000, Arduino MICRO/Genuino MICRO, Arduino 101/Genuino 101, Arduino Gemma, entre otros y luego están para sistemas embebidos: Arduino Yún y Arduino Tian que vienen con las características de un Arduino Leonardo en cuanto a microcontrolador con puertos de entradas/salidas (digitales/analógicas) y con un micróprocesador Atheros con sistema operativo Linino (GNU/Linux OpenWrt reducido) para propósitos generales en la misma placa (lo mejor de ambos mundos).

Arduino cuenta además con placas de expansión (shields): Arduino GSM Shield, Arduino Ethernet Shield, Arduino WiFi Shield, Arduino Wireless SD Shield, Arduino USB Host Shield, Arduino Motor Shield, Arduino Wireless Proto Shield, Arduino Proto Shield.

A continuación se menciona en forma resumida a las placas Arduino investigadas con más detalle: Arduino Uno, Arduino Yún y Arduino Tian.

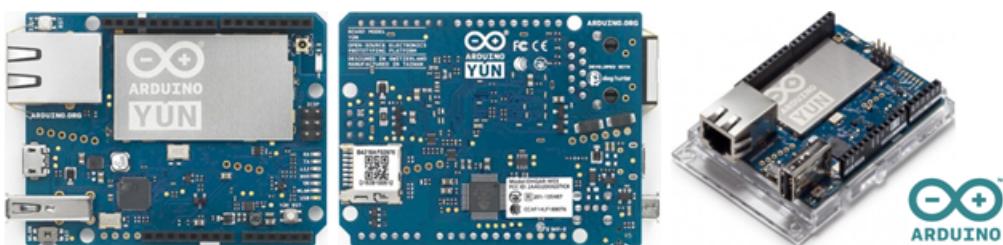
- *Arduino Uno*: según Arduino (2019) es una placa de microcontrolador basada en ATmega328P. Tiene 14 pines de entrada/salida digital (de los cuales 6 pueden usarse como salidas PWM), 6 entradas analógicas, un cristal de cuarzo de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP y un botón de reinicio. Contiene todo lo necesario para soportar el microcontrolador; simplemente conéctelo a una computadora con un cable USB o enciéndalo con un adaptador de CA a CC o una batería para comenzar. "Uno" significa uno en italiano y fue elegido para marcar el lanzamiento de Arduino Software (IDE) 1.0. La placa Uno y la versión 1.0 de Arduino Software (IDE) fueron las versiones de referencia de Arduino, ahora evolucionadas a versiones más nuevas. La placa Uno es la primera de una serie de placas Arduino USB, y el modelo de referencia para la plataforma Arduino.

Figura 25: Arduino Uno. Fuente: Arduino (2019)



- *Arduino Yún*: según Arduino (2019) es la placa perfecta para usar al diseñar dispositivos conectados y, más en general, proyectos de Internet de las cosas. Combina el poder de Linux con la facilidad de uso de Arduino. Arduino Yún es una placa de microcontrolador basada en ATmega32u4 y Atheros AR9331. El procesador Atheros admite una distribución de Linux basada en OpenWrt llamada Linino OS. La placa tiene soporte incorporado para Ethernet y WiFi, un puerto USB-A, ranura para tarjeta micro-SD, 20 pines de entrada/salida digital (7 de ellos pueden usarse como salidas PWM y 12 como entradas analógicas), un cristal de 16 MHz oscilador, una conexión micro USB, un encabezado ICSP y 3 botones de reinicio. El Yún se distingue de otras placas Arduino por su capacidad de comunicarse con la distribución de Linux a bordo, ofreciendo una poderosa computadora en red con la facilidad de un Arduino. Además de los comandos de Linux como cURL, puede escribir sus propios scripts de shell y python para interacciones robustas. El Yún es similar al Leonardo con el ATmega32u4, excepto que tiene Linux a bordo. (tiene comunicación USB incorporada, eliminando la necesidad de un procesador secundario).

Figura 26: Arduino Yún. Fuente: Arduino (2019)



- *Arduino Tian*: según Arduino (2019) el Arduino Tian libera el poder de una pequeña computadora con Linux conectada a un microcontrolador de 32 bits, está alimentada por la MCU SAMD21 de Atmel, que presenta un núcleo ARM Cortex® M0+ de 32 bits y un Qualcomm Atheros AR9342, que es un procesador MIPS altamente integrado que funciona hasta 533 MHz y un Módulo Wi-Fi de doble banda con IEEE 802.11n 2x2 2.4/5 GHz rico en funciones. Qualcomm Atheros MIPS admite una distribución de Linux, basada en OpenWRT llamada Linino. El Arduino Tian también tiene una construcción en memoria eMMC de 4 GB que puede ser útil para construir proyectos. Es posible encender/apagar el puerto Linux desde la MCU para reducir el consumo de energía.

Figura 27: Arduino Tian. Fuente: Arduino (2019)



Raspberry Pi (micro computador). Según Pi (2019) es una placa de computadora de bajo costo del tamaño de una tarjeta de crédito que se conecta a un monitor de computadora o TV, y utiliza un teclado y ratón estándar, es un pequeño dispositivo capaz de permitir a las personas de todas las edades explorar la informática y aprender a programar en lenguajes como Scratch y Python, fue desarrollada por la Fundación Raspberry Pi en Caldecote, Reino Unido. Raspberry Pi es capaz de hacer todo lo que esperaría que hiciera una computadora de escritorio, desde navegar por Internet y reproducir videos de alta definición, hasta hacer hojas de cálculo, procesamiento de textos y juegos. Además, la Raspberry Pi tiene la capacidad de interactuar con el mundo exterior y se ha utilizado en una amplia gama de proyectos de creadores digitales, desde máquinas de música y detectores de padres hasta estaciones meteorológicas y twitteando pajareras con cámaras infrarrojas. En esencia, el Raspberry Pi es una placa de un tamaño minúsculo (cercano al de una tarjeta de crédito). Posee un micro procesador ARM con potencia de hasta 1GHz, integrado en un chip Broadcom BCM2835. Además cuenta con 512 MB de RAM, un GPU Videocore IV, todo lo necesario para poder ejecutar programas básicos, navegar por internet y por supuesto programar.

En cuanto a software, la Raspberry Pi soporta numerosos sistemas operativos(SO), mayoritariamente basados en el núcleo Linux. Por defecto, trae instalada en la tarjeta de memoria SD el sistema operativo libre Raspbian, que está basado en Debian y está optimizado para el hardware de este microcomputador. La fundación da soporte para las descargas de este SO y de otros como RISC OS 5, Pidora (basado en Fedora) y Arch Linux ARM (derivado de Arch Linux). También soporta, entre otros, Android, Firefox OS, Debian, OpenWrt y Slackware Linux.

Modelos actuales de Raspberry Pi son: Raspberry Pi Compute Module, Raspberry Pi Compute Module 3, Raspberry Pi Compute Module 3+, Raspberry Pi Zero, Raspberry Pi Zero W, Raspberry Pi Model A+, Raspberry Pi Model 3A+, Raspberry Pi Model B+, Raspberry Pi 2 Model B, Raspberry Pi 2 v1.2, Raspberry Pi 3 Model B, Raspberry Pi 3 Model B+ y Raspberry Pi 4.

A continuación se comenta la Raspberry Pi 3 Model B, dicha placa es muy utilizada en la actualidad.

- *Raspberry Pi 3 Model B:* de acuerdo a Pi (2019) es el primer modelo de la Raspberry Pi de tercera generación, es una computadora de placa única con LAN inalámbrica y

conectividad Bluetooth. Reemplazó al Raspberry Pi 2 Modelo B en febrero de 2016. Teniendo las siguientes especificaciones: CPU Quadcom 1.2MHz Broadcom BCM2837 64bit, 1 GB de RAM, BCM43438 LAN inalámbrica y Bluetooth de baja energía (BLE) a bordo 100 Base Ethernet, GPIO extendido de 40 pines, 4 puertos USB 2.0, salida estéreo de 4 polos y puerto de video compuesto, HDMI de tamaño completo, puerto de cámara CSI para conectar una cámara Raspberry Pi, puerto de pantalla DSI para conectar una pantalla táctil Raspberry Pi, puerto Micro SD para cargar un sistema operativo y almacenar datos, fuente de alimentación micro USB conmutada actualizada hasta 2.5A.

Figura 28: Raspberry Pi 3 Model B. Fuente: Pi (2019)



Beagleboard.org (placa de desarrollo). De acuerdo a BeagleBoard.org (2018) todos los diseños son de código abierto y los componentes están disponibles para que cualquiera pueda fabricar hardware compatible. Solicitamos contacto y permiso antes de considerar el uso del nombre BeagleBoard.org en cualquier producto.

Las placas son computadoras de una sola placa de bajo costo y sin ventilador basadas en procesadores Texas Instruments de baja potencia que cuentan con el núcleo de la serie ARM Cortex-A con toda la capacidad de expansión de las máquinas de escritorio actuales, pero sin el volumen, los gastos o el ruido. Inicialmente, el desarrollo estaba dirigido a habilitar las distribuciones de Linux para mejorar el soporte para dispositivos ARM. Con un enorme éxito y el apoyo de numerosas distribuciones de Linux, el desarrollo se ha centrado más en habilitar la informática física simplificada en dispositivos avanzados habilitados para GUI y/o en red con una experiencia de aprendizaje lista para usar y soporte súper simple para entornos de desarrollo familiares a casi todos los desarrolladores, desde Ubuntu, QNX, Windows Embedded, Android y herramientas web hasta Bare-Metal e incluso programación de estilo Arduino/Wiring.

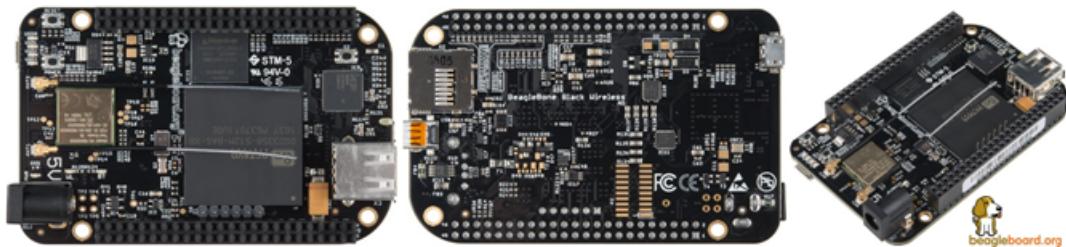
Modelos actuales de Beagleboard.org son: PocketBeagle, BeagleBone Blue, BeagleBone Black, BeagleBone AI, BeagleBone Black Wireless, SeeedStudio BeagleBone Green, SeeedStudio BeagleBone Green Wireless, SanCloud BeagleBone Enhanced, BeagleBoard-xM y BeagleBoard, además cuenta con módulos adicionales: PocketBeagle TechLab Cape, PocketBeagle GamePup Cape, BeagleBone Comms Cape, BeagleBone Load Cape, BeagleBone Motor Cape, BeagleBone Power Cape, BeagleBone Proto Cape, BeagleBone Relay Cape, BeagleBone Robotics Cape.

A continuación se describe la placa BeagleBone Black Wireless para propósitos generales como plataforma de desarrollo.

- *BeagleBone Black Wireless*: de acuerdo a es una plataforma de desarrollo de bajo costo y apoyada por la comunidad para desarrolladores y aficionados. El BeagleBone usa el microporcesador Sitara AM3358 de Texas Instruments, un chip procesador ARM que funciona a 1 GHz. Si desea realizar operaciones en tiempo real, el procesador ARM de BeagleBone no funcionará bien ya que Linux no es un sistema operativo en tiempo real. Sin embargo, el chip Sitara contiene dos microcontroladores de 32 bits, llamados unidades programables en tiempo real o PRU. Al usar una PRU, se puede lograr un control rápido, determinista y en tiempo real de los pines y dispositivos de E/S. Con respecto a la conectividad reemplaza el puerto Ethernet 10/100 con 802.11 b/g/n a bordo de Wi-Fi y Bluetooth de 2.4 GHz. Aprovechando una asociación con Octavo Systems y diseñado en CadSoft Eagle, BeagleBone Black Wireless es la computadora IoT Linux del tamaño de una tarjeta de crédito más fácil de usar y modificar.

En referencia a la compatibilidad software, la BeagleBone Black Wireless trae instalada en la memoria flash la distribución libre Ångström Linux, y que está soportada por la fundación. Pero no es el único SO que soporta. Se puede formatear la memoria flash e instalar Debian, Android, Ubuntu, Gentoo, Minix o ArchLinux entre otros, aunque no están soportados por BeagleBoard.org.

Figura 29: Beaglebone Black Wireless. Fuente: BeagleBoard.org (2018)



Comparativas de placas bases. Se realizaron varias tablas comparativas teniendo en cuenta detalles del microcontrolador, microporcesador, interfaces y datos generales de las placas bases embebidas estudiadas.

Tabla 12: Comparativo Microcontrolador. Fuente: Autor

Microcontrolador				
Marca	Arduino	Arduino	Raspberry Pi	Beaglebone
Modelo	Yún	Tian	3 Model B	Black Wireless
Microcontrolador	ATmega32u4	SAMD21G18	No	Embebido AM3358
Arquitectura	Microchip AVR 8-bit	ARM Cortex-M0+	No	2xPRU 32-bit
Voltaje de Operación	5V	3,3 V	No	3,3 V
Memoria Flash	32 KB (4 KB utilizado por bootloader)	256 KB	No	4 GB eMMC
SRAM	2.5 KB	32 KB	No	No
Velocidad de Reloj	16 MHz	48 MHz	No	200 MHz
EEPROM	1 KB	No	No	32 KB
I/O total	20	20	40	2x46 (P8 y P9) (92)
Digital (max)	20	20	26	25
PWM (max)	7	12	4	7
Analog (max)	12 (10-bit de resolución)	6	No	7
Corriente DC por pines I/O	40 mA (I/O pines) 50 mA (3.3V pin)	7 mA (I/O pines)	16 mA	No se especifica

Tabla 13: Comparativo Microporcesador. Fuente: Autor

Microporcesador				
Marca	Arduino	Arduino	Raspberry Pi	Beaglebone
Modelo	Yún	Tian	3 Model B	Black Wireless
Procesador	Atheros AR9331	Atheros AR9342	Broadcom BCM2837	Sitara AM3358BZCZ100
Arquitectura	MIPS	MIPS	64-bit quad-core ARMv8	ARM Cortex A8
Voltaje de operación	3.3 V	3.3 V	3.3 V	3.3 V
Memoria Flash	16 MB	16 MB + 4 GB eMMC	No	4 GB eMMC
RAM	64 MB DDR2	64 MB DDR2	1 GB DDR2	512 MB DDR3
Velocidad de reloj	400 MHz	560 MHz	1,2 GHz	1 GHZ

Tabla 14: Comparativo Interfaces. Fuente: Autor

Interfaces				
Marca	Arduino	Arduino	Raspberry Pi	Beagleboard.org
Modelo	Yún	Tian	3 Model B	Beaglebone Black Wireless
Serial	1xUART, 1xSPI, 1xTWI	1xUART, 1xSPI, 1xTWI	1xUART, 2xSPI, 2xTWI	6xUART, 1xSPI, 1xTWI
USB	2x2.0 (1xhost y 1xcliente)	2x2.0 (1xhost y 1xcliente)	4x2.0 (4xhost)	5x2.0 (4xhost y 1xcliente)
Bluetooth	No	CSR8510, Bluetooth with EDR / BLE 4.0	Bluetooth 4.2 BLE	Bluetooth 4.1 BLE
Wi-Fi	802.11 b/g/n 2.4GHz	802.11 b/g/n 2.4 GHz dual-band	2.4 GHz y 5 GHz IEEE 802.11.b/g/n/ac	WiLink 1835 802.11 b/g/n de 2,4 GHz
Ethernet	802.3 10/100 Mbit/s	802.3 10/100/1000 Mbit/s	Gigabit Ethernet sobre USB 2.0 (max. 300 Mbps)	802.3 10/100/1000 Mbit/s
GPU	No	No	Broadcom VideoCore IV	SGX530 3D, 20M Polygons/S
Entrada video	No	No	MIPI CSI (picamara)	No
Salida video	No	No	RCA (PAL y NTSC), HDMI, MIPI DSI	microHDMI
Salida audio	No	No	conector 3,5 mm y HDMI	HDMI
Lector de Tarjeta	microSD	No	microSD	microSD

Tabla 15: Comparativo General. Fuente: Autor

General				
Marca	Arduino	Arduino	Raspberry Pi	Beaglebone
Modelo	Yún	Tian	3 Model B	Black Wireless
Voltaje entrada	5 V (USB, ext. u opcional PoE)	5 V (USB)	5 V (USB)	5 V (USB o DC power)
Consumo de energía	250 mA	470 mA	2,5 A	1,0 A
Dimensiones	53 mm x 68.5 mm	53 mm x 68.5 mm	85.60 mm x 56.5 mm	126 mm x 76 mm
Peso	34 g	36 g	42 g	91 g
Software abierto	SI	SI	SI	SI
Hardware abierto	SI	SI	NO	SI
IDE	SI (Arduino IDE)	SI (Arduino IDE)	NO	SI (Cloud9 IDE + nodejs + javascript)
Sistema Operativo	Linino (embebido Linux OpenWrt)	Linino (embebido Linux OpenWrt)	Raspbian, Fedora, Arch Linux, Kano OS, Windows IoT Core, OpenWrt, entre otros	Linux (embedido Amstrong), Android, Windows Embedded CE, QNX, ThreadX
Tipo	Placa de desarrollo	Placa de desarrollo	Micro-PC	Placa de desarrollo - Micro PC
Precio	Gs. 500.000	Gs. 650.000	Gs. 480.000	Gs. 600.000

Módulos pre-actuadores para accionar al actuador.

Relé electromecánico. De acuerdo a Iberorobotic (2019) es especial para prototipado rápido con Arduino u otros controladores, muy fácil de conectar y programar, sin necesidad de librerías específicas de control, sólo necesita el voltaje de salida de un pin (TTL- 5 V) de una placa Arduino o similar para controlar su funcionamiento, con diodo LED indicador de estado. Este módulo relé está especialmente indicado para prototipado rápido de proyectos de electrónica, domótica y robótica que necesiten actuar sobre grandes cargas como motores, bombas, climatizadores, iluminación, o cualquier otro tipo de instalación o maquinaria. Puede ser controlado directamente por una amplia gama de microcontroladores tales como Arduino, 8051, AVR, PIC, DSP, ARM, MSP430, etc. con nivel de salida TTL (5V). Los módulos de relé vienen por canales de: 1, 2, 3, 4 ... hasta 18.

Relé de estado sólido (SSR). Según Embajadores (2018) este es una placa con un relé de estado sólido (OMRON G3MB-202P) basado en optotriac de activación paso por cero, permite atacar cargas de 240 V y con un consumo de hasta 2 A, a diferencia de un relé electromecánico, este tipo de relé permite conectar cargas con elevadas capacidades de conmutación, puesto que no son mecánicos. Sin embargo este relé de estado sólido no es válido para conmutar por ejemplo tensiones de corriente continua, como se trata de un relé de estado sólido la carga se encuentra optoaislada de la señal de activación. Los módulos de relé vienen por canales de: 1, 2, 3, 4 ... hasta 18 también.

Comparativos de módulos relé de un canal. Se realizaron comparativas teniendo en cuenta detalles de los pre-actuadores del tipo relé de un canal.

Figura 30: Pre-Actuadores Songle, Tongling y Omron respectivamente. Fuente:Iberorobotic (2019), Embajadores (2018)



Tabla 16: Comparativo de módulos de relé de un canal. Fuente: Autor

Tipo	Electromecánico	Electromecánico	Estado sólido
Voltaje control	5 VCD	5 VCD	4 a 6 VDC
Componente Relé	Songle SRD-05VDC-SL-C	Tongling JQC-3FF-S-Z	OMRON G3MB-202P-DC5
Corriente activación	20 mA	20 mA	1 mA
Tensión máxima AC	250 V / 10 A	250 V / 10 A	240 V / 2 A
Tensión máxima DC	30 V / 10 A	30 V / 10 A	No soporta
LEDs indicadores	2	2	No posee
Voltaje de Disparo	Bajo	Alto/Bajo	Bajo
Peso	30 gr.	38 gr.	40 gr.
Tamaño	43 x 30 x 15 mm	50 x 22 x 20 mm	55 x 33 x 20 mm
Pines interface	Pin 1: +5V (VCC) Pin 2: Entrada disparo Pin 3: Tierra (GND)	Pin 1: +5V (VCC) Pin 2: Tierra (GND) Pin 3: Entrada disparo	Pin 1: +5V (VCC) Pin 2: Tierra (GND) Pin 3: Entrada disparo
Pines salida relé	NC: normalmente cerrado COM: común NO: normalmente abierto	NC: normalmente cerrado COM: común NO: normalmente abierto	NO: normalmente abierto COM: común
Precio	Gs. 35.000	Gs. 40.000	Gs. 80.000

Módulos actuadores para accionar el riego.

Electroválvula de corriente alterna. De acuerdo a Mechatronics (2019) sirve para controlar el flujo de agua en una tubería, esta válvula solenoide es un tipo de electroválvula todo/nada o abierto/cerrado, tienen dos partes: el solenoide y el cuerpo metálico, el solenoide es un electroimán que al ser energizado se desplaza junto con el diafragma de la válvula y permite el paso del fluido, en esta válvula el electroimán actúa directamente sobre el diafragma por lo que no necesita una presión mínima de funcionamiento, la válvula se mantiene abierta mientras el solenoide está energizado con 240 VAC, cuando no está alimentado un resorte se encarga de regresar la válvula a su posición de reposo, que en este caso es del tipo normalmente cerrada (NC). Para su control se puede utilizar un microcontrolador como Arduino, ESP8266 o un Raspberry Pi. Es necesario utilizar un driver de potencia entre el microcontrolador y la válvula, en este caso se puede usar un módulo relay o SSR.

Electroválvula de corriente continua. Según Mechatronics (2019) esta válvula solenoide es ideal para proyectos de domótica, automatización industrial, riego automatizado, agricultura de precisión. Con voltaje de operación de 12 VDC normalmente cerrado de rosca externa de 1/2" de tipo diafragma.

Comparativos de electroválvulas para agua. . .

Figura 31: Electroválvulas para agua AC-DC respectivamente. Fuente: Mechatronics (2019)

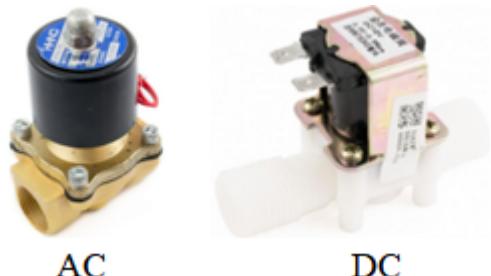


Tabla 17: Comparativos de electroválvulas para agua AC-DC. Fuente: Autor

Tipo	Corriente Alterna (AC)	Corriente Continua (CC)
Voltaje operación	220 VAC (+/- 10% tolerancia)	12 VDC
Corriente operación	36.36 mA	600 mA
Modelo	AirTac 2W-200-20 AC220	Genérico
Potencia consumo	8 W	8 W
Temperatura funcionamiento	(-5°) C a 80° C	5° C a 100° C
Presión funcionamiento mínima	0 Bar	0,2 Bar
Presión funcionamiento máximo	10 Bar	8 Bar
Tiempo de respuesta (apertura)	No especifica	≤ 0.15 s
Tiempo de respuesta (cerrado)	No especifica	≤ 0.3 s
Conector tubería	Rosca interna 3/4" NPT hembra	Rosca externa 1/2" NPS macho
Reposo	Normalmente cerrado	Normalmente cerrado
Tipo de válvula	Diafragma	Diafragma
Recomendación	Todo tipo de fluidos	Fluidos de baja viscosidad
Dimensiones	52,5 x 85,6 x 32,2 mm	60 x 85 x 26 mm
Precio	Gs. 240.000	Gs. 60.000

Módulos sensores para temperatura y humedad relativas.

DHT11. Según Mechatronics (2019) es un sensor digital de temperatura y humedad relativa de bajo costo y fácil uso. Integra un sensor capacitivo de humedad y un termistor para medir el aire circundante, y muestra los datos mediante una señal digital en el pin de datos (no posee salida analógica). Utilizado en aplicaciones académicas relacionadas al control automático de temperatura, aire acondicionado, monitoreo ambiental en agricultura y más. Utilizar el sensor DHT11 con las plataformas Arduino/Raspberry Pi/Nodemcu es muy sencillo tanto a nivel de software como hardware. A nivel de software se dispone de librerías para Arduino con soporte para el protocolo "Single bus". En cuanto al hardware, solo es necesario conectar el pin VCC de alimentación a 3-5V, el pin GND a Tierra (0V) y el pin de datos a un pin digital.

DHT21. Según Mechatronics (2019) es un sensor digital de temperatura y humedad relativa de buena precisión en un empaque robusto. Integra un sensor capacitivo de humedad, un termistor y un microcontrolador encargado de realizar la conversión analógica a digital. Su empaque de plástico es más robusto comparado a los sensores DHT11 y DHT22, esto hace del DHT21 un sensor ideal para aplicaciones en exteriores. Utilizado en aplicaciones de control automático de temperatura, aire acondicionado, monitoreo ambiental en agricultura y más. Utilizar el sensor DHT21 con las plataformas Arduino, Raspberry Pi, Nodemcu es muy sencillo tanto a nivel de software como hardware. A nivel de software se dispone de librerías para Arduino con soporte para el protocolo "Single bus". En cuanto al hardware, solo es necesario conectar el cable de color Rojo a la fuente de alimentación 3-5V, el cable Negro a Tierra (0V) y el cable Amarillo a un pin digital en nuestro Arduino. Si se desea conectar varios sensores DHT21 a un mismo Arduino, cada sensor debe tener su propio pin de datos. Quizá la única desventaja del sensor es que sólo se puede obtener nuevos datos cada 2 segundos. Cada sensor es calibrado en fábrica para obtener unos coeficientes de calibración grabados en su memoria OTP, asegurando alta estabilidad y fiabilidad a lo largo del tiempo. El protocolo de comunicación entre el sensor y

el microcontrolador emplea un único hilo o cable, la distancia máxima recomendable de longitud de cable es de 20m., de preferencia utilizar cable apantallado. Proteger el sensor de la luz directa del sol (radiación UV).

DHT22. Según Mechatronics (2019) el DHT22 (AM2302) es un sensor digital de temperatura y humedad relativa de buen rendimiento y bajo costo. Integra un sensor capacitivo de humedad y un termistor para medir el aire circundante, y muestra los datos mediante una señal digital en el pin de datos (no posee salida analógica). Utilizado en aplicaciones de control automático de temperatura, aire acondicionado, monitoreo ambiental en agricultura y más. Utilizar el sensor DHT22 con las plataformas Arduino, Raspberry Pi, Nodemcu es muy sencillo tanto a nivel de software como hardware. A nivel de software se dispone de librerías para Arduino con soporte para el protocolo "Single bus". En cuanto al hardware, solo es necesario conectar el pin VCC de alimentación a 3-5V, el pin GND a Tierra (0V) y el pin de datos a un pin digital. Si se desea conectar varios sensores DHT22 a un mismo Arduino, cada sensor debe tener su propio pin de datos. Quizá la única desventaja del sensor es que sólo se puede obtener nuevos datos cada 2 segundos. Cada sensor es calibrado en fabrica para obtener unos coeficientes de calibración grabados en su memoria OTP, asegurando alta estabilidad y fiabilidad a lo largo del tiempo. El protocolo de comunicación entre el sensor y el microcontrolador emplea un único hilo o cable, la distancia máxima recomendable de longitud de cable es de 20m., de preferencia utilizar cable apantallado. Proteger el sensor de la luz directa del sol (radiación UV).

Comparativos de sensores de temperatura y humedad relativa. .

Figura 32: Módulos de sensores de temperatura y humedad relativa. Fuente: Mechatronics (2019)



Tabla 18: Comparativos de sensores de temperatura y humedad relativa. Fuente: Autor

Modelo	DHT11	DHT21 (AM2311A)	DHT22 (AM2302)
Fabricante	OSEPP Electronics	AOSONG	AOSONG
Voltaje operación	3V - 5V DC	3,5 V - 5,5 V DC	3 V - 6 V DC
Consumo Energía	2,5mA	0,5mA	2 mA
Rango medición temperatura	0 a 50 °C	(-40) a 80 °C	(-40) a 80 °C
Precisión medición temperatura	± 2,0 °C	± 0,5 °C	<± 0,5 °C
Resolución temperatura	0,1 °C	0,1 °C	0,1 °C
Rango medición humedad	20 % a 90% RH.	0 % a 100 % RH.	0 % a 100 % RH.
Precisión medición humedad	±5 % RH.	±3 % RH.	±2 % RH.
Resolución humedad	1 % RH.	0,1 % RH.	0,1 % RH.
Tiempo muestreo	1 seg.	2 seg.	2 seg.
Interface digital	Single-bus (bidireccional)	Single-bus (bidireccional)	Single-bus (bidireccional)
Dimensiones	16 x 12 x 5 mm	60 x 28 x 13mm	20 x 15 x 8 mm
Peso	1 gr.	17 gr	3 gr
Carcasa	plástico celeste	plástico negro	plástico blanco
Pines/cable (vista frontal)	Pin 1: +5V (VCC) Pin 2. Datos (DATA) Pin 3. No usado (NC) Pin 4. Tierra (GND)	Cable rojo: +5V (VCC) Cable negro: Tierra (GND) Cable amarillo: Datos digitales I/O	Pin 1: +5V (VCC) Pin 2. Datos (DATA) Pin 3. No usado (NC) Pin 4. Tierra (GND)
Precio	Gs. 20.000	Gs. 70.000	Gs. 50.000

Módulos sensores para humedad del suelo.

Adafruit Stemma. Según Electronics (2019) mediante el sensor de suelo Adafruit STEMMA (sensor de humedad capacitivo I2C) las mediciones capacitivas usan solo una sonda, no tienen metal expuesto y no introducen corrientes de CC a las plantas, utiliza el sistema de medición táctil capacitiva incorporado en el chip ATSAMD10D14A, que obtiene una lectura que oscila entre 200 (muy seco) y 2000 (muy húmedo), como beneficio adicional, obtiene también la temperatura ambiente desde el sensor de temperatura interno en el microcontrolador, no es de alta precisión, quizás sea bueno para + o - 2 grados Celsius. Para que se pueda usar el sensor con casi cualquier microcontrolador, tenemos una interfaz I2C, se conecta un cable JST-PH de 4 pines al microcontrolador o computadora de placa única a 3 - 5 V de potencia, tierra, I2C SDA e I2C SCL para leer la temperatura y la capacidad medición de acuerdo al código de programa creado.

YL-69/YL-38. De acuerdo a Mechatronics (2019) este sensor utiliza 2 electrodos que pasan corriente a través del suelo, la resistencia entre los electrodos dependerá de la humedad del suelo, por lo que para un suelo muy húmedo tendremos una resistencia muy baja (corto circuito) y para un suelo muy seco la resistencia será muy alta (circuito abierto). El electrodo (YL-69) va conectado a una tarjeta de acondicionamiento (YL-38) en la cual un OpAmp (LM393) en modo comparador entrega una salida digital y otra analógica. La salida digital se activa cuando el nivel de humedad es menor al deseado, este nivel se puede regular con el potenciómetro de la tarjeta. La salida analógica nos entrega un voltaje desde 0V para un suelo muy húmedo hasta 5V para un suelo muy seco, este sensor es ideal para controlar el nivel de humedad de plantas o realizar un sistema totalmente automatizado de riego.

KS0049. Según Didactic (2019) es un sensor simple de humedad del suelo que tiene como objetivo detectar la humedad del suelo. Si el suelo no tiene agua, el valor analógico emitido por el sensor disminuirá, de lo contrario, aumentará. El sensor se configura con dos sondas insertadas en el suelo, luego con la corriente atraviesa el suelo, el sensor obtendrá valor de resistencia al leer los cambios actuales entre las dos sondas y convertir dicho valor de resistencia en contenido de humedad. Cuanto mayor sea la humedad (menos resistencia), mayor será la conductividad del suelo. La superficie del sensor se ha sometido a un proceso de metalización para prolongar su vida útil. Insértalo en el suelo y luego usa el convertidor AD para leerlo. Con la ayuda de este sensor, la planta puede recordarte: necesito agua.

Comparativos de sensores de humedad de suelo. . .

Figura 33: Sensores de humedad del suelo. Fuente: Didactic (2019); Electronics (2019); Mechatronics (2019)

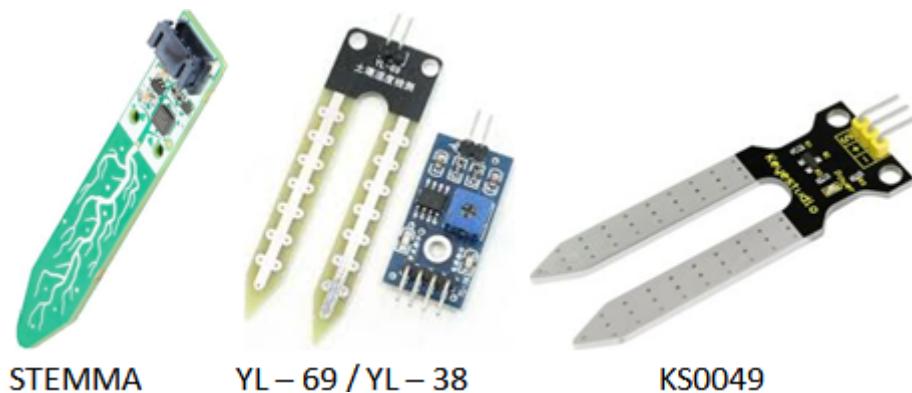


Tabla 19: Comparativo de sensores humedad del suelo. Fuente: Autor

Modelo	STEMMA	YL-69 / YL-38	KS0049
Fabricante	Adafruit	SEN-HUS	Keyes
Voltaje operación	3 V - 5 V DC	3,3 V - 5 V DC	3,3 V - 5 V DC
Energía consumo	40 mA	35 mA	< 20 mA
Rango medición analógica	200 (seco) 2000 (muy húmedo)	0 V (muy húmedo) 5 V (seco)	0 V (seco) 2,3 V (muy húmedo)
Tipo	Capacitivo (I2C)	Resistivo (comparativo)	Resistivo (comparativo)
Dimensiones	76,2 x 14 x 7 mm	60 x 20 x 5 mm	20 x 60 mm
Peso	10 gr.	20 gr.	12 gr.
Sonda	1	2	2
Superficie sonda	Estaño	Estaño	Pintura dorada
Pines/cable (vista frontal)	Pin 1: VCC Pin 2: Tierra (GND) Pin 3: I2C SDA Pin 4: I2C SCL	Pin 1: A0 (analógico) Pin 2: D0 (digital) Pin 3: Tierra (GND) Pin 4: VCC	Pin 1: Datos (DATA) Pin 2: +5V (VCC) Pin 3: Tierra (GND)
Precio	Gs. 50.000	Gs. 35.000	Gs. 40.000

Módulos sensores para detección de lluvia.

MH-RD / YL-38. Según Electrónica (2014) este sensor está pensado para el control de dispositivos en función de la lluvia. La placa cuenta con una superficie niquelada para resistir mejor la oxidación y alargar su vida útil. Consiste en una PCB MH-RD (tipo YL-83) y un módulo YL-38 que contiene un CI comparador LM393 SMD muy estable, un led de encendido y otro de activación de salida digital. Este último cuenta con 2 pines de conexión hacia el módulo MH-RD, 2 pines para la alimentación y 2 pines de datos. El sensor de lluvia detecta el agua que completa los circuitos en las pistas impresos de sus tarjetas de sensores. La tarjeta del sensor actúa como una resistencia variable que cambiará de 100 KOhms cuando está mojado a 2 MOhms cuando está seco. En resumen, cuanto más húmedo sea el tablero, mayor será la corriente que se llevará a cabo. Ideal para producir dispositivos relacionados con invernaderos, huertas y aquellos se vean afectados en función del clima. Además, cuenta con sensibilidad ajustable por medio del potenciómetro digital azul.

Octopus EF04024. Según Caldas (2013) este sensor de lluvia/vapor de ElecFreaks utiliza la medida análoga, se puede conectar directamente al escudo de expansión Arduino I/O. El voltaje de salida aumentará cuando aumente la humedad de la superficie del sensor. Incluso se podría usar como panel táctil.

Keyes T1592. De acuerdo Electrónica (2014) es un sensor de identificación y detección de gotas de lluvia fácil de usar, compacto, liviano, de bajo consumo de energía, alta sensibilidad. El principio es medir el tamaño de la cantidad de traza de gotas de agua a través de la línea con una serie de pistas paralelas expuestas, basada en los valores analógicos de salida del sensor, además se puede conectar directamente por ejemplo: Arduino, microcontrolador STC, microcontrolador AVR, etc.

Comparativos de sensores de detección de lluvia.

Figura 34: Sensores de detección de lluvia. Fuente: Caldas (2013); Electrónica (2014)



Tabla 20: Comparativo de sensores de detección de lluvia. Fuente: Autor

Modelo	MH - RD / YL-38	Octopus EF04024	T1592
Fabricante	SEN-HUS	ElecFreaks	Keyes
Voltaje operación	3,3 V - 5 V DC	3,3 V - 5 V DC	5 VDC
Energía consumo	35 mA	35 mA	< 20 mA
Rango medición analógica	0 V (lluvioso) 5 V (soleado)	0 (soleado) 950 (lluvioso)	0 V (soleado) 5 V (lluvioso)
Tipo	Resistivo (comparativo)	Resistivo	Resistivo
Dimensiones Sensor	59 x 20 x 5 mm	38,2 x 27 x 4,4 mm	66 x 20 x 8 mm
Área de detección	54 x 40 mm	3,14 x 19,7 x 19,7 mm	41 x 16 mm
Peso	20 gr.	10 gr.	3 gr.
Superficie detección	Estaño - Plomo	Estaño - Plomo	Estaño - Plomo
Pines/cable (vista frontal)	Pin 1: A0 (analógico) Pin 2: D0 (digital) Pin 3: Tierra (GND) Pin 4: VCC	Pin 1: Tierra (GND) Pin 2: +5V (VCC) Pin 3: Datos (DATA)	Pin 1: Datos (DATA) Pin 2: +5V (VCC) Pin 3: Tierra (GND)
Precio	Gs. 35.000	Gs. 15.000	Gs. 20.000

Módulos sensores para detección de luz.

BH1750. Según Mechatronics (2019) el módulo BH1750 es un sensor de iluminación digital para medición de flujo luminoso (iluminancia) de la empresa Rohm Semiconductor. Posee un conversor interno de 16-bit, por lo que entrega una salida digital en formato I2C. Su desempeño es mejor al de un Foto-Resistor (LDR), pues no es necesario realizar conversiones de voltaje para obtener datos interpretables. El BH1750 entrega la intensidad luminosa directamente en unidades Lux (Lx). El lux es la unidad derivada del S.I. de unidades para la iluminancia o nivel de iluminación. Equivale a un lumen/m². Se usa en la fotometría como medida de la luminancia, tomando en cuenta las diferentes longitudes de onda según la función de luminosidad, un modelo estándar de la sensibilidad a la luz del ojo humano.

CdS LDR GL5528. Según Mechatronics (2019) medir la cantidad de la luz ya no será un problema con la ayuda de este sensor LDR. Una fotorresistencia o LDR es un dispositivo (CdS: sulfuro de cadmio) cuya resistencia varía de acuerdo con la cantidad de luz que reciba, son muy útiles para proyectos de control automático de iluminación, seguidores de luz solar, interruptores crepusculares, entre otros. Para poder utilizarlo se debe agregar una resistencia de 10K ohm en serie y conectarlo a una entrada analógica (A0-A5) del Arduino como un divisor de tensión.

MOD-LDR / YL-38. De acuerdo Mechatronics (2019) este módulo posee 2 salidas, una analógica que se debe conectar a una entrada analógica y así utilizar el ADC (conversor analógico digital). La salida digital posee solo 2 estados: activo/apagado, el cambio de un estado a otro depende del umbral que se fije con el potenciómetro del módulo. La salida digital puede utilizarse para controlar un relay y así realizar una acción dependiente de la intensidad de luz.

Comparativos de sensores para detección de luz.

Figura 35: Sensores para detección de luz. Fuente: Mechatronics (2019)



Tabla 21: Comparativos de sensores para detección de luz. Fuente: Autor

Modelo	BH1750	CdS LDR GL5528	MOD-LDR / YL38
Fabricante	Rohm Semiconductor	Nanyang Senba Optical Electronic	SEN-HUS
Voltaje operación	3 V - 5 V DC	max 150 V DC	3,3 V - 5 V DC
Energía consumo	7 mA	max 670 mA	35 mA
Rango medición	1 - 65535 lux	0 - 100 lux	0 V (lluvioso) 5 V (soleado)
Tipo	Digital (I2C)	Resistivo	Resistivo (comparativo)
Dimensiones Sensor	18,6 x 14,5 mm	5 x 3 mm	54 x 40 mm
Peso	1,1 gr.	0,3 gr.	20 gr.
Pines/cable (vista frontal)	Pin 1: ADDR Pin 2: SDA Pin 3: SCL Pin 4: Tierra (GND) Pin 5: VCC	Pin 1: Resistencia + Tierra (GND) Pin 2: +5V (VCC)	Pin 1: A0 (analógico) Pin 2: D0 (digital) Pin 3: Tierra (GND) Pin 4: VCC
Precio	Gs. 35.000	Gs. 2.000	Gs. 35.000

Módulos sensores para medir flujo del líquido (caudalímetro). Los modelos caudalímetros estudiados corresponden a los de tipo electrónico a turbina, existen otros de tipo: mecánico, ultrasónico y electromagnético.

FS300A. De acuerdo a Mechatronics (2019) el sensor de flujo de agua de 3/4" FS300A sirve para medir caudal de agua en tuberías de 3/4" de diámetro. También puede ser empleado con otros líquidos de baja viscosidad, como: bebidas gasificadas, bebidas alcohólicas, combustible, etc. Es un caudalímetro electrónico de tipo turbina. Compatible con sistemas digitales como Arduino, Pic, Raspberry Pi, PLCs. El sensor posee tres cables: rojo (VCC: 5VDC), negro (tierra) y amarillo (salida de pulsos del sensor de efecto Hall). El funcionamiento del sensor es de la siguiente forma: el caudal de agua ingresa al sensor y hace girar una turbina, la turbina está unida a un imán que activa un sensor de efecto Hall, que a su vez emite un pulso eléctrico que puede ser leído por la entrada digital de un Arduino o PLC. El sensor de efecto Hall está aislado del agua, de manera que siempre se mantiene seco y seguro. Como el volumen de agua por cada pulso es fijo y de un valor conocido (promedio) podemos contar la cantidad de pulsos por unidad de tiempo (segundo o minuto), luego multiplicar el valor de volumen/pulso por la cantidad de

pulsos y así determinar el caudal o flujo de agua. Se recomienda utilizar interrupciones por hardware en el Arduino para detectar o contar los pulsos del sensor. Tenga en cuenta que este no es un sensor de precisión por lo que la orientación, presión del agua y otras condiciones pueden afectar la medición. Se recomienda calibrar el sensor realizando mediciones con volúmenes conocidos. Calibrado puede llegar a tener una precisión de hasta 10%. Fórmula: Flujo del agua en L/min = Pulsaciones del sensor (Hz) / 5.5

YF-S201. Según Mechatronics (2019) el sensor de flujo de agua de 1/2" YF-S201 sirve para medir caudal de agua en tuberías de 1/2" de diámetro. Sus aplicaciones y funcionamiento es igual al modelo FS300A. Fórmula: Flujo del agua en L/min = Pulsaciones del sensor (Hz) / 7.5

FS400A. Teniendo en cuenta a Mechatronics (2019) el sensor de flujo de agua de 1" FS400A sirve para medir caudal de agua en tuberías de 1" de diámetro. Sus aplicaciones y funcionamiento es igual al modelo FS300A. Fórmula: Flujo del agua en L/min = Pulsaciones del sensor (Hz) / 4.8

Comparativos de sensores para medir flujo del líquido .

Figura 36: Sensores para medir flujo del líquido. Fuente: Mechatronics (2019)



Tabla 22: Comparativos de sensores para medir flujo del líquido. Fuente: Autor

Modelo	FS300A	YF-S201	FS400A G1
Fabricante	Meinte	Sea (YIFA)	Meinte
Voltaje operación	5V - 24V DC	5V - 18V DC	5V - 24V DC
Energía consumo	15 mA	15 mA	15 mA
Capacidad de carga	10mA (5VDC)	10mA (5VDC)	10mA (5VDC)
Salida	Onda cuadrada pulsante	Onda cuadrada pulsante	Onda cuadrada pulsante
Rango de Flujo	1-60L/min	1-30L/min	1-60L/min
Volumen promedio por pulso	3,03mL	2,25mL	3,47mL
Pulsos por litro	330	450	280
Factor de conversión	5,5	7,5	4,8
Rosca externa	3/4" NPS	1/2" NPS	1" NPS
Presión de trabajo	1,2MPa (12bar)	1,75MPa (17bar)	1,2Mpa (12bar)
Temperatura funcionamiento	(-)25°C a 80°C	(-)25°C a 80°C	(-)25°C a 80°C
Presición	(+/-)10%	(+/-)10%	(+/-)10%
Material	Plástico color negro	Plástico color negro	Plástico color negro
Dimensiones Sensor	63 x 36 mm	63 x 35 mm	70 x 42 mm
Pines/cable (vista frontal)	Pin 1: +5VDC (rojo) Pin 2: GND (negro) Pin 3: PWM (amarillo)	Pin 1: +5VDC (rojo) Pin 2: GND (negro) Pin 3: PWM (amarillo)	Pin 1: +5VDC (rojo) Pin 2: GND (negro) Pin 3: PWM (amarillo)
Precio	Gs. 90.000	Gs. 80.000	Gs. 120.000

Módulos de reloj de tiempo real (RTC).

RTC DS1307. De acuerdo a Mechatronics (2019) el módulo está basado en el RTC (Real Time Clock) DS1307 de MAXIM y la EEPROM AT24C32 de ATMEL. Ambos circuitos integrados comparten el mismo bus comunicación con el Protocolo I2C. La memoria EEPROM AT24C32 permite almacenar 32Kbits (4K Bytes) de datos de manera permanente. Funciones del RTC: hora (segundos, minutos, horas), fecha(número día, día de la semana, mes y año), compensación de año bisiesto, pin de salida a 1 Hz para llevar el tiempo en otros componentes. Al reverso tiene un socket para insertar una batería de 3V, que permite al RTC mantener en funcionamiento la hora y fecha cuando se apague el Arduino o se le quiete la alimentación principal. Esta batería depende de la marca y del tiempo apagado, puede durar desde los 3 hasta los 10 años aproximadamente.

DS3231MPMB1. Según Electronics (2019) el módulo periférico DS3231MPMB1 proporciona el hardware necesario para interconectar el reloj de tiempo real (RTC) DS3231M con cualquier sistema que utilice puertos de expansión compatibles con Pmod configurables para la comunicación I²C. El IC es un RTC I²C de bajo costo y extremadamente preciso. El dispositivo incorpora una entrada de batería y mantiene un cronometraje preciso cuando se interrumpe la alimentación principal del dispositivo. La integración del resonador de sistemas microelectromecánicos (MEMS) mejora la precisión a largo plazo del dispositivo y reduce el recuento de piezas en una línea de fabricación.

BOB-14558. De acuerdo a Electronics (2019) el módulo SparkFun Qwiic RTC (BOB-14558) es un módulo RTC, basado en el chipset RV-1805, diseñado para el ecosistema Qwiic Connect. El módulo RTC consume una potencia ultra baja, funciona a solo 22 nA, utilizando un supercondensador para energía de respaldo. Este supercondensador puede cargar y descargar ciclos sin ninguna degradación de la batería. El módulo RTC utiliza el práctico sistema Qwiic y se comunica a través de I2C, lo que facilita la obtención de las lecturas. El chip RV-1805 incorporado en este módulo RTC viene con dos osciladores internos, un cristal de diapasón de 32.768 kHz y un oscilador basado en RC de baja potencia. Estos osciladores pueden cambiar automáticamente entre dos utilizando el cristal preciso que corrige el oscilador RC cada pocos minutos. Esto permite que el módulo mantenga una fecha y hora muy precisas durante un año.

Comparativo de módulos de reloj de tiempo real (RTC). .

Figura 37: Módulos de reloj de tiempo real (RTC). Fuente: Electronics (2019); Mechatronics (2019)



Tabla 23: Comparativos de módulos de reloj de tiempo real (RTC). Fuente: Autor

Modelo	RTC DS1307	DS3231MPMB1	BOB-14558
Fabricante	Elecrow	Maxim Integrated	SparkFun Electronics
Voltaje operación	3,3 V - 5 V DC	2,3 V - 5,5 V DC	1,6 V - 3,6 V
Energía consumo	500 nA	300 uA	22nA
Rango temperatura operación	(-45) a 85 °C	(-45) a 85 °C	(-40) a 85 °C
Tipo batería	CR1225 3 V Li-Ion	CR-1025/BN 3V Dióxido de litio y manganeso	CR927 3 V Li-Ion
Precisión tiempo	± 23.0 ppm	± 2.0 ppm	± 2.0 ppm
Tipo	Digital (I2C)	Digital (I2C)	Digital (I2C)
Dimensiones	27 x 28 x 8,4 mm	3,0 x 29,7 x 21,0 mm	26 x 26 x 4 mm
Peso	10 gr.	30 gr.	10 gr.
Pines/cable (vista frontal)	Pin 1: DS (Señal Digital) Pin 2: SCL Pin 3: SDA Pin 4: VCC Pin 5: Tierra (GND)	Pin 1: Reset Pin 2: DS (Señal Digital) Pin 3: SCL Pin 4: SDA Pin 5: Tierra (GND) Pin 6: VCC	Pin 1: Tierra (GND) Pin 2: 3,3 V Pin 3: SDA Pin 4: SCL Pin 5: PSW Pin 6: DS (Señal Digital) Pin 7: Reset
Precio	Gs. 35.000	Gs. 120.000	Gs. 110.000

Periféricos de indicadores luminosos (salida).

LED-01. De acuerdo a Electronics (2019) los leds agregan siempre un toque llamativo al proyecto. Este modelo de led tiene la cubierta difusa, por lo que la luz se difumina por casi todo el led, logrando un efecto mas agradable a la vista, es ideal para trabajar con Arduino y Protoboard, estos leds están disponibles en 5 colores: Blanco, Azul, Rojo, Verde y Ámbar, se puede utilizar como indicadores, para carteles, cubos led y un sin fin de aplicaciones.

LED-RGB-B5C. Teniendo en cuenta a Electronics (2019) los Leds RGB (Red, Green, Blue) integran en un solo dispositivo 3 leds de los colores: Rojo (Red), Verde (Green) y Azul (Blue), para obtener todos los colores de este led es necesario utilizar PWM de Arduino, el color a mostrar dependerá del duty cycle que se asigne a cada canal. Cada led necesita de 3 salidas tipo pwm, un Arduino Uno entrega 6 salidas PWM permitiendo controlar 2 leds RGB simultáneamente, este modelo de led rgb es de 5mm de diámetro y tiene la configuración cátodo común (negativo común). Se construir diversos proyectos, como: cubos led, carteles luminosos, lámparas multicolores, indicadores, etc.

LED-MATRIZ-MAX7219. Según Electronics (2019) al trabajar con leds tarde o temprano se necesita una matriz led, si bien es posible construir una matriz utilizando leds individuales, también podemos optar por una matriz de fábrica ahorrando mucho tiempo, de forma similar la circuitería necesaria para manejar nuestra matriz puede ser realizada a partir de transistores discretos o utilizar un chip especializado como el MAX7219. Este módulo integra una matriz de leds de 8x8 y un chip MAX7219, lo que permite trabajar rápidamente con matrices led y utilizando solo 4 cables para la comunicación. Estos módulos se pueden conectar

en cascada utilizando un solo bus SPI y mostrar textos o gráficos más grandes. El chip MAX7219 es magnífico pues permite controlar 64 leds de forma independiente (o 8 dígitos de 7-segmentos), incluye un decoder BCD, posee una memoria RAM estática interna de 8x8. Permite el control del brillo de forma digital y analógica. Solo necesita una resistencia externa para su funcionamiento. Se maneja tan sólo con 3 pines y es compatible con SPI, QSPI, y MICROWIRE, un dato adicional es que se puede colocar en serie varios módulos del mismo.

Comparativos de indicadores luminosos. .

Figura 38: Indicadores luminosos. Fuente: Electronics (2019)

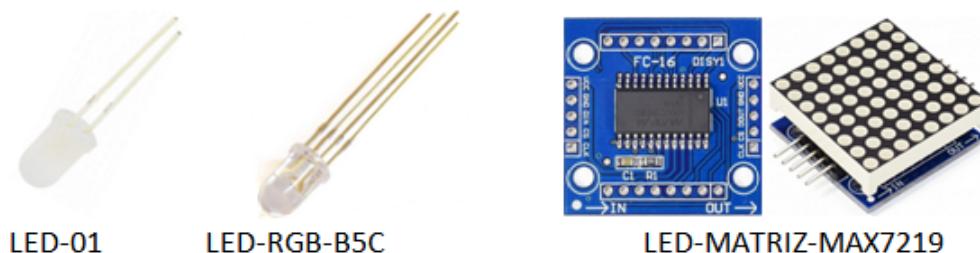


Tabla 24: Comparativo indicadores luminosos. Fuente: Autor

Modelo	LED-01	LED-RGB-B5C	LED-MATRIZ-MX7219
Voltaje operación	Rojo: 1,8 V Verde: 2,25 V Azul: 3,0 V	Rojo: 1,8 a 2,2 V Verde: 3,0 - 3,4 V Azul: 3,0 -3,4 V	5 V DC
Energía consumo	20 mA	30 mA	100 mA
Rango temp. Oper.	(-40) a 70 °C	(-25) a 85 °C	(-40) a 85 °C
Tipo	1 x Difuso	3 x Ultrabrilante Cátodo común	Hiper Rojo / Matriz 8 x 8 Cátodo común
Angulo visión	20°	60°	120°
Luminosidad	Rojo: 800 mcd Verde: 4000 mcd Azul: 900 mcd	Rojo: 800 mcd Verde: 4000 mcd Azul: 900 mcd	Rojo: 60 mcd
Diámetro/ Dimensión	5 mm	5 mm	32 x 32 mm
Peso	0,1 gr.	0,4 gr.	10 gr.
Pines/cable (vista frontal)	Pin 1: Cátodo (GND) Pin 2: Ánodo (VCC)	Pin 1: Ánodo Rojo Pin 2: Cátodo (GND) Pin 3: Ánodo Verde Pin 4: Ánodo Azul	Pin 1: VCC Pin 2: Tierra (GND) Pin 3: Digital (IN) Pin 4: CS Pin 5: CLK
Precio	Gs. 250	Gs. 2.000	Gs. 30.000

Periféricos de tarjetas de memorias de almacenamiento de datos (entrada/salida). En este apartado se analizan tarjetas de memoria de 16 GB de almacenamiento del tipo microSD debido a que las placas bases estudiadas lo soportan. Teniendo en cuenta a Association (2019) los estándares SD permiten a los fabricantes ofrecer productos de alto rendimiento que mejoran la experiencia de millones de personas cada día que escuchan música, graban videos, toman fotos, guardan datos y usan un teléfono móvil.

La memoria flash NAND de los microSD está formada por muchas celdas que contienen bits, y esos bits se activan o desactivan mediante una carga eléctrica, la forma en que se organizan esas celdas de encendido/apagado representa los datos almacenados en el SSD. El número de bits en esas celdas también determina el nombre del flash, por ejemplo, el flash de celda de nivel único (SLC) contiene un solo bit en cada celda. Los tipos de NAND Flash son: SLC (Single Level Cell), eMLC (Enterprise Multi-Level Cell), MLC (Multi-Level Cell) y TLC (Triple-Level Cell).

Comparativo de tarjetas de memorias de almacenamiento de datos.

Figura 39: Tarjetas de memorias de almacenamiento de datos. Fuente: Electronics (2019)



Tabla 25: Comparativo tarjetas de memorias de almacenamiento de datos. Fuente: Autor

Modelo	SDCS/16GB	SDSDQAF3-008G-I	SFSD8192N3BM1TO-E-LF-2CP-STD
Fabricante	Kingston	Sandisk	Swissbit
Tamaño memoria	16 GB	16 GB	16 GB
Familia SD	SDHC	SDHC	SDHC
Tipo NAND Flash	MLC	MLC	MLC
Clase velocidad	10	10	10
Clase velocidad UHS	U1	U1	U1
Clase velocidad video	No	No	No
Lectura sostenida	80 MB/s	80 MB/s	48 MB/s
Escritura sostenida	10 MB/s	50 MB/s	48 MB/s
Voltaje operación	3,3 V DC	2,7 a 3,6 V DC	2,7 a 3,3 V DC
Energía consumo	100 mA	115 mA	100 mA
Temperatura operativa	(-25) a 85 °C	(-25) a 85 °C	(-40) a 85 °C
Dimensiones Sensor	15 x 11 x 1 mm	15 x 11 x 1 mm	15 x 11 x 1 mm
Nivel de bloqueo flash	20.000	20.000	20.000
Precio	Gs. 65.000	Gs. 60.000	Gs. 170.000

Periféricos de video captura web (entrada).

Comparativos de cámaras web. En base a páginas web de fabricantes se realiza un cuadro comparativo de cámaras web según especificaciones técnicas obtenidas de marcas reconocidas.

Figura 40: Cámaras web. Fuente: Logitech (2019); Microsoft (2019); Razer (2019)



Tabla 26: Comparativo de cámaras web. Fuente: Autor

Modelo	C270	Kiyo	LifeCam Studio
Fabricante	Logitech	Razer	Microsoft
Tipo conexión	USB 2.0	USB 2.0	USB 2.0
Resolución imagen	3 megapixeles	4 megapixeles	5 megapixeles
Fotograma por segundos (fps)	30	30	30
Resolución de video	720p	1080p	720p
Micrófono integrado	Sí	Sí	Sí
Campo de visión	60 grados	81,6 grados	60 grados
Longitud cable	1,5 metros	1,5 metros	1,5 metros
Fuente	Logitech (2019)	Razer (2019)	Microsoft (2019)
Precio	Gs. 150.000	Gs. 1.100.000	Gs. 210.000

Fuentes de alimentación de energía. En este apartado la fuente de alimentación de energía son las del tipo adaptador AC/DC de 100-240 V de entrada alterna y con salida del tipo micro USB 5V DC a 1 A como mínimo, que son empleadas en promedio en las placas bases estudiadas, atendiendo por el consumo que implica agregar módulos adicionales a las mismas y que las fuentes de alimentación de energía basadas en baterías/pilas de cualquier material no son aconsejables para proyectos destinados a un servicio continuo (24/7) de control.

Comparativos de fuentes de alimentación de energía. .

Figura 41: Fuentes de alimentación de energía. Fuente: Electronics (2019); Mechatronics (2019)



Tabla 27: Comparativo de fuentes de alimentación de energía. Fuente: Autor

Modelo	UBP-008	HN-5258	SGA12E05USB
Tipo	Conmutada	Conmutada	
Voltaje entrada	110-240 V (CA) (50/60 Hz)	100-240 V (CA) (50/60 Hz)	90-264 V (CA) (50/60 Hz)
Voltaje salida	5 V (CC)	5 V (CC)	6 V (CC)
Corriente nominal	2 (A)	2,5 (A)	30
Potencia suministrada	10 W	12,5 W	12 W
Conecotor salida CC	USB A hembra	Micro USB B macho	USB A hembra
Indicador	No	No	No
Eficiencia	75%	80%	80%
Protección	Cortocircuito, sobrecorriente y sobrecarga	Cortocircuito, sobrecorriente y sobrecarga	Cortocircuito, sobrecorriente y sobrecarga
Longitud cable	No posee	1 metro con interruptor encendido / apagado	No posee
Temperatura operacional	(-10) a 60 °C	(-20) a 60 °C	(-20) a 60 °C
Fuente	Mechatronics (2019)	Mechatronics (2019)	Electronics (2019)
Precio	Gs. 40.000	Gs. 75.000	Gs. 80.000

Aspectos Legales

Ley N° 3239. De los Recursos Hídricos del Paraguay. República del Paraguay (2007)

Capítulo I: Objetivo **Artículo 1º.**- La presente Ley tiene por objeto regular la gestión sustentable e integral de todas las aguas y los territorios que la producen, cualquiera sea su ubicación, estado físico o su ocurrencia natural dentro del territorio paraguayo, con el fin de hacerla social, económica y ambientalmente sustentable para las personas que habitan el territorio de la República del Paraguay.

Capítulo II: Principios **Artículo 3º.**- La gestión integral y sustentable de los recursos hídricos del Paraguay se regirá por los siguientes Principios:

c) Los recursos hídricos poseen usos y funciones múltiples y tal característica deberá ser adecuadamente atendida, respetando el ciclo hidrológico, y favoreciendo siempre en primera instancia el uso para consumo de la población humana.

Artículo 4º.- La Política Nacional de los Recursos Hídricos se abocará a los siguientes objetivos básicos:

a) Impulsar el uso sustentable, racional e integral de los recursos hídricos, como elemento condicionante de la supervivencia del género humano y de todo el sistema ecológico, promoviendo con amplio sentido protecciónista su mejor disfrute, el de los otros recursos naturales y del ambiente. Para ello, deberá tenerse en cuenta la unidad del recurso en cualquiera de las etapas del ciclo hidrológico, la inter dependencia entre los distintos recursos naturales y entre los distintos usos del agua, el condicionamiento del ambiente, la protección, conservación y restauración de territorios productores de agua controlando y manejando las influencias que es capaz de producir la acción humana.

c) Impulsar y mantener un adecuado conocimiento integral de los recursos hídricos en cuanto a cantidad, calidad y oportunidad en su aprovechamiento, así como de su carácter condicionante de las actividades humanas, dinamizando la investigación científica, sistemática, operativa y tecnológica, a través o en colaboración con los organismos competentes.

f) Impulsar el aprovechamiento de los recursos hídricos en forma racional y conforme a un adecuado ordenamiento jerárquico de los valores, usos esenciales, socio-económicos e individuales a satisfacer. Para ello es conveniente fijar las prioridades vitales y aquellos criterios que han de aplicarse para definir un orden objetivo para jerarquizar los otros usos, según las circunstancias que determinen la selección de las respectivas demandas, evaluadas por las características regionales y dentro del complejo de las políticas contenidas en esta Ley y la política general.

g) Propender al uso múltiple de los recursos hídricos y a la integración coordinada desde el punto de vista funcional entre todos ellos mediante el manejo racional y administración común a toda manifestación hídrica, asignando valor prioritario a los proyectos de usos múltiples

sobre los de uso singular, siempre que ello esté justificando técnica, social y económicamente y minimicen los efectos o daños al ambiente.

h) Tender a la economía en el uso de los recursos hídricos, a través de su utilización racional y eficiente, posibilitando, así la disponibilidad para otros usos, previendo sobre su derroche, contaminación y degradación.

k) Promover en el seno de la sociedad el conocimiento de los métodos y tecnologías necesarias para el adecuado manejo, uso y conservación de los recursos hídricos, en atención a que ellos, más que cualquier otro recurso natural, están destinados al uso de todos.

ll) Disponer la revisión integral de la legislación y las reglamentaciones existentes y mantener su permanente actualidad, con el fin de armonizarlas con los tratados y convenios internacionales ratificados por nuestro país, adecuar su comprensión, mejorar su alcance y simplificar su aplicación, ello fundamentalmente, en cuanto al conocimiento y aprovechamiento de los recursos hídricos a través de la aplicación de la ciencia, la técnica y la tecnología que resulten apropiadas, para promover e impulsar un conveniente desarrollo del sector.

Capítulo III: Definiciones Artículo 5º.- Para los efectos de esta Ley, los siguientes conceptos tendrán los significados que se expresan a continuación:

e) Agua superficial: Agua o recurso hídrico que permanece o escurre sobre la superficie de la tierra, en estado líquido o sólido.

f) Agua: Compuesto componente del ambiente y recurso natural relativamente renovable formado por dos partes de hidrógeno y diecisésis partes de oxígeno en peso.

x) Recursos hídricos: Comprende el total de las aguas superficiales, subterráneas, atmosféricas, y agua útil generada por tecnologías nuevas tales como: aguas desalinizadas, regeneradas y otras, en sus diferentes estados físicos, incluidos sus cauces, lechos, álveos y acuíferos y que pueden ser utilizadas de alguna forma en beneficio del hombre.

aa) Uso para fines domésticos: Es la utilización de los recursos hídricos destinados exclusivamente a la satisfacción de necesidades de núcleos familiares humanos en los límites necesarios a la alimentación, a los cuidados de la higiene, al lavado y a la producción agraria básica para el consumo del núcleo familiar.

Capítulo VI: Derechos de Uso y Aprovechamiento de los Recursos Hídricos Artículo 13.- Todo habitante de la República del Paraguay es sujeto de derecho de uso y aprovechamiento de los recursos hídricos con diversos fines, en armonía con las normas, prioridades y limitaciones establecidas en la presente Ley, con excepción a lo establecido en la Ley N° 1614/00 «GENERAL DEL MARCO REGULATORIO Y TARIFARIO DEL SERVICIO PÚBLICO DE PROVISIÓN DE AGUA POTABLE Y ALCANTARILLADO SANITARIO PARA LA REPÚBLICA DEL PARAGUAY».

Artículo 15.- Los recursos hídricos superficiales y subterráneos de uso para fines domésticos y de producción familiar básica que sean utilizados de manera directa por el usuario, sin

intermediación de ningún tipo, son de libre disponibilidad, no están sujetos a permisos ni concesiones ni impuestos de ningún tipo y deberán estar inscriptos en el Registro Nacional de Uso y Aprovechamiento de los Recursos Hídricos, al solo fin de su contabilización en el Balance Hídrico Nacional. Se reglamentará el control de este tipo de uso.

Artículo 18.- Será prioritario el uso y aprovechamiento de los recursos hídricos superficiales y subterráneos para consumo humano. Los demás usos y aprovechamiento seguirán el siguiente orden de prioridad:

- a) Satisfacción de las necesidades de los ecosistemas acuáticos.
- c) Uso y aprovechamiento para actividades agropecuarias, incluida la acuacultura.

Ley N° 716. Que sanciona delitos contra el medio ambiente. (República del Paraguay, 1996)

Artículo 5º.- Serán sancionados con penitenciaría de uno a cinco años y multa de 500 (quinientos) a 1.500 (mil quinientos) jornales mínimos legales para actividades diversas no especificadas:

e) Los que eludan las obligaciones legales referentes a medidas de mitigación de impacto ambiental o ejecuten deficientemente las mismas.

Artículo 14º.- Se consideran agravantes:

- a) El fin comercial de los hechos;

El Plan Nacional de Desarrollo Paraguay 2030 (PND 2030)

D. Marco estratégico

Líneas transversales **Sostenibilidad ambiental:** Disminuir los desequilibrios ambientales propios de la actividad económica y los asentamientos humanos. Significa mejorar considerablemente la calidad de vida de la población en asentamientos humanos, con viviendas mejoradas, acceso universal al agua potable, a servicios básicos de saneamiento y a sistemas de gestión de residuos; también disminuir los niveles de contaminación ambiental de las industrias, con planificación y control sobre las zonas de explotación de recursos naturales, reduciendo así la pérdida del patrimonio natural y de la biodiversidad nativa. Además, implica la creación de oportunidades para incentivar la protección y recuperación de los ecosistemas y remunerar los servicios ambientales.

Instala como idea central de la visión a largo plazo la utilización sostenible de la tierra, el agua y los bosques, por ello, promueve la adecuada regulación del uso extractivo de los recursos naturales.

Proyecto de Intervención

Objetivo General

Construir un dispositivo con indicadores de lluvia, temperatura ambiente, humedad del suelo y luz solar para configurar y supervisar en forma remota la automatización del proceso de riego por goteo de un huerto urbano en macetas.

Objetivos Específicos

- Ensamblar las partes necesarias del hardware del dispositivo.
- Desarrollar, instalar y configurar el software para el dispositivo.
- Acceder en forma remota para configurar y supervisar al dispositivo.

Justificación

Los dispositivos electrónicos para el proceso de riego por goteo de un huerto urbano en macetas en el mercado actual tienen funcionalidades limitadas y sobre todo no son automatizados con la ayuda de indicadores ambientales combinados como lluvia, temperatura ambiente, humedad del suelo y luz solar, ni tener en cuenta el volumen de capacidad de la maceta, ni caudal del gotero, además no disponen acceso remoto al mismo para configurar y supervisar el estado en tiempo real de los indicadores ambientales, ni registrar eventos, ni notificar alertas mediante correo electrónico, ni de visualizar el ambiente físico a través de una cámara web (video/foto).

Por lo manifestado contribuirá al mercado de tecnologías verdes o ecológicas, debido a que mediante ésta adaptativa tecnológica se pretenderá racionalizar, cuantificar, supervisar y cuidar el uso adecuado del vital líquido: el agua, a más de ahorrar tiempo y dinero a largo plazo en el mantenimiento de una zona verde como es un huerto urbano con macetas en su proceso de riego por goteo.

La descripción del producto o situación esperada

El dispositivo a nivel de hardware seguirá el diagrama en bloque de un sistema de control automatizado (unidad de control, sensores, pre-actuador, actuador y proceso) se compondrá de cuatro cajas plásticas externas separadas pero unidas mediante conectores especiales.

* Primera caja: alojará a la unidad de control consistente en una placa base embebida de fuente abierta y hardware abierto, como conexión intermedia entre la unidad de control y los demás módulos del dispositivo, además se empleará una placa de circuito impreso (PCB) creado en forma artesanal para concentrar y ordenar las derivaciones de los pines digitales/analógicos y de suministros de energía con la ayuda de cables del tipo Dupont, teniendo los siguientes módulos conectados;

- Sensores de: detección de lluvia, temperatura ambiente/humedad relativa, humedad del suelo y detección de luz,

- Módulo de reloj de tiempo real que mantendrá la fecha y hora del sistema ante cortes de energía eléctrica, y

- Indicadores luminosos para los estados de la unidad de control (rojo: en espera, azul: acceso remoto, verde: riego activo, turquesa: fuera de horario de riego y violeta: dispositivo desactivado o sin configuración activa)

Y por el puerto USB disponible en la placa base embebida se conectará la cámara web con el cual se podrá visualizar en tiempo real el ambiente físico circundante al huerto urbano.

* Segunda caja: constará de los elementos transductores de los sensores que estarán situados en una maceta de referencia, conectados mediante cables par trenzado a la unidad de control.

* Tercera caja: constará del pre-actuador, módulo de un canal de relé que será el mecanismo que activará/desactivará al actuador para el proceso de riego, además de una luz piloto para indicar que se encuentra encendido (Led rojo), ésta caja

* Cuarta caja: contendrá al actuador que será una electroválvula (solenoide) de corriente continua normalmente cerrada (NC), la cual estará conectada a la unidad de control mediante un empalme de retorno (cable cordón) para activación/desactivación proveniente del pre-actuador, el actuador estará ubicado entre en el mecanismo de suministro de agua y el micro tubo de riego, el cual será el encargado de activar/desactivar el proceso de riego como una llave de paso (abierto/cerrado), en la salida de la electroválvula irá conectado el sensor de caudal de agua .

En relación a las fuentes de alimentación de energía: tanto para la unidad de control como para el actuador se utilizarán fuentes adecuadas y estables según los requerimientos de cada módulo.

El dispositivo a nivel de software constará de un firmware con el algoritmo de control de automatización de la unidad de control, programas externos para funciones específicas como estrategia de control de riego a partir de una estructura de almacenamiento de datos, servicio de telemetría, notificación de alertas, activación de una cámara web y un software multiplataforma de arquitectura cliente/servidor que actuará como interfaz gráfica del usuario.

Las proporciones de cliente y/o mercado potencial

Se ha detectado un creciente y atractivo interés del mercado en la implementación de huertos urbanos como actividad recreativa y comprensión sobre la importancia de entender, valorar y respetar el cuidado de la naturaleza, así como la alternativa sostenible para la producción de verduras frescas que conlleva a la satisfacción por consumir nuestros propios cultivos, sin olvidar que cada día las empresas dedicadas a la fabricación de macetas mejoran las tecnologías para maximizar la cantidad de recipientes en espacios reducidos mediante muebles esféricos o verticales, por lo mencionado el dispositivo construido con todas sus características facilitará

el suministro de agua en forma ecológica a ésta marcada tendencia y así preservar este tipo de zona verde.

La proyección financiera que se espera con la implementación del proyecto

Tabla 28: Proyección financiera. Fuente: Autor

	ITEMS	Sub Total
1	Placa Base Embebida Arduino Yún	500.000
2	Sensor de detección de lluvia (MH/YL-38)	40.000
3	Sensor de temperatura ambiente/ humedad relativa (DHT22)	50.000
4	Sensor de humedad del suelo (YL-69/YL-38)	35.000
5	Sensor de luz (CdS LDR GL5528)	2.000
6	Cámara web (Logitech c270)	150.000
7	Tarjeta de memoria microUSB (SandDisk)	60.000
8	Módulo de reloj de tiempo real (RTC DS1307)	35.000
9	Módulo de relé de un canal (electromecánico)	40.000
10	Electroválvula	60.000
11	Sensor de caudal (caudalímetro YF-S201)	80.000
12	Placa circuito virgen 7 x 10 cmm	15.000
13	Resistencia de 10 K ohm	1.250
14	Resistencias de 220 ohm	1.500
15	Led RGB (B5C)	5.000
16	Cables varios	30.000
17	Cajas plásticas externas	50.000
18	Adaptador AC/DC 100-240 V 50/60 Hz (CA) - Micro USB 5 VCC 10 W	40.000
19	Adaptador AC/DC 100-240 V 50/60 Hz (CA) - 12 VCC 5 W	40.000
20	Desarrollo del software	500.000
21	Instalación y configuración	100.000
TOTAL GENERAL Gs.		1.834.750

Los involucrados del proyecto, con sus respectivas responsabilidades

Lic. Gustavo Eloy Alcaráz Bogado – Director y desarrollador del proyecto

Descripción de las Etapas

El proyecto a partir de este momento adquiere en nombre de **SistRiego** (Sistema de Riego) para facilitar las documentaciones de las etapas desarrolladas para la construcción del dispositivo.

Etapa 1: Inicial.

Exploración de bases teóricas. Referentes al ambiente físico de intervención (huerto urbano, cultivo en macetas y riego por goteo), sistemas de control y automatización, fundamentos de la electricidad, electrónica e informática, transferencia de video en tiempo real y sistemas embebidos o empotrados.

Búsqueda de referencias técnicas. Sobre placas bases embebidas, módulos: pre-actuadores (relé de un canal), actuadores (electroválvula), sensores (lluvia, temperatura ambiente/humedad relativa, humedad del suelo, luz y caudal de agua), módulos de reloj de tiempo real, tarjetas de memoria microSD, cámaras web y fuentes de alimentación de energía.

Elección y obtención del hardware para el dispositivo. A partir de las referencias técnicas se realizarán comparaciones entre alternativas según requerimientos del dispositivo.

Etapa 2: Hardware del dispositivo.

Elaboración del esquema eléctrico entre módulos. Según montaje en el tablero de pruebas (Protoboard) entre la placa base y sus módulos independientes adquiridos mediante cables de tipo dupont.

Creación de placa de circuito impreso de conexión entre módulos. Partiendo del esquema eléctrico elaborado crear; las pistas del circuito impreso (PCB) correspondientes, imprimir en papel térmico, termo transferir a la placa virgen de cobre, sumergir en ácido para remoción del cobre no deseado, posteriormente pulir las pistas resultantes, perforar la placa de circuito impreso en los orificios correspondientes para los componentes, por último adherir y plastificar la impresión del circuito eléctrico completo con descripciones de los componentes, módulos y pines en la cara opuesta de las pistas de cobre de la placa.

Ensamblado de los módulos. Con la placa de circuito impreso creado proceder a la soldadura de los zócalos de conexión con cables unifiliares del tipo dupont macho-hembra y componentes, interconectar entre sí las partes del dispositivo: unidad de control, módulos, sensores y pre-actuador.

Etapa 3: Software del dispositivo.

Elaboración del flujo de control automatizado. Mediante diagramas de flujos representar gráficamente el funcionamiento integral del dispositivo en cuanto a la obtención de datos de los sensores (telemetría) y la estrategia de control para el pre-actuador/actuador, además de peticiones adicionales y comportamiento de los indicadores lumínicos por estados del dispositivo.

Instalación y configuración de aplicativos para la placa base embebida. En este tópico se instalarán y configurarán los aplicativos necesarios para comunicar, preparar y utilizar la placa base embebida de acuerdo a los requerimientos del dispositivo propuesto.

Creación de una estructura de almacenamiento de datos. En tal sentido se debe consignar la manera mas eficiente de guardar y recuperar datos de una manera bien estructura para el dispositivo.

Programación del firmware de control automatizado. Una vez obtenido los diagramas de flujos de control automatizado se crearan las rutinas del firmware para configuración inicial y del ciclo repetitivo del dispositivo, una vez terminado se enviarán a la memoria flash del microcontrolador del dispositivo para su operativa mediante conexión USB, en caso necesario se iterará este proceso para verificar y optimizar el firmware.

Creación del programa externo para el servicio de telemetría. Se deberá tener un programa externo al firmware que pueda correr en segundo plano en S.O. de Linux OpenWrt como servicio de acceso de datos de telemetría (datos de sensores).

Creación del programa externo para el servicio de notificación de alertas. Se deberá tener un programa externo al firmware que pueda correr en segundo plano en S.O. de Linux OpenWrt como servicio notificación de alertas del dispositivo por vía de correo electrónico.

Creación del programa externo de estrategia de control de riego. Este programa será el encargado de comparar los datos obtenidos de los sensores del dispositivo, deberán ser enviados éstos datos como argumentos y contrastados con datos obtenidos mediante consulta a una estructura de almacenamiento datos creada, de donde se obtendrá los parámetros de referencia de la especie de cultivo en cuanto a valores de rangos mínimos y máximos permitidos de humedad del suelo, temperatura ambiente e intensidad luminosa además de otros campos en la configuración activa (cantidad de agua al día, tiempo activo, tiempo espera, caudal del gotero, cantidad de macetas, etc.), una vez comparado consultará el último registro de riego en el día

para enviar al dispositivo la respuesta de activar/desactivar el proceso de riego según cantidad de agua suministrada contra el tope estipulado por la especie.

Creación del programa externo de activación de la cámara web. Se deberá tener un programa externo al firmware que pueda activar la cámara web conectada al puerto USB de la placa de Arduino Yún y mediante aplicativos instalados en S.O. de Linux OpenWrt puedan capturar una imagen o transmitir video en tiempo real a la interfaz gráfica del usuario.

Creación del software de interfaz gráfica del usuario. Culminado los servicios en segundo plano se procederá al desarrollo de la interfaz del usuario para interactuar con el dispositivo mediante una aplicación de plataforma web que estará alojado en la tarjeta de almacenamiento microSD del dispositivo, una vez desplegado los usuarios podrán acceder a la misma mediante una conexión wifi directa al dispositivo o en forma indirecta mediante acceso inalámbrico de la red ya sea por un ruteador (router) o un punto de acceso (AP) del usuario previa configuración del equipo de red del usuario con el dispositivo construido.

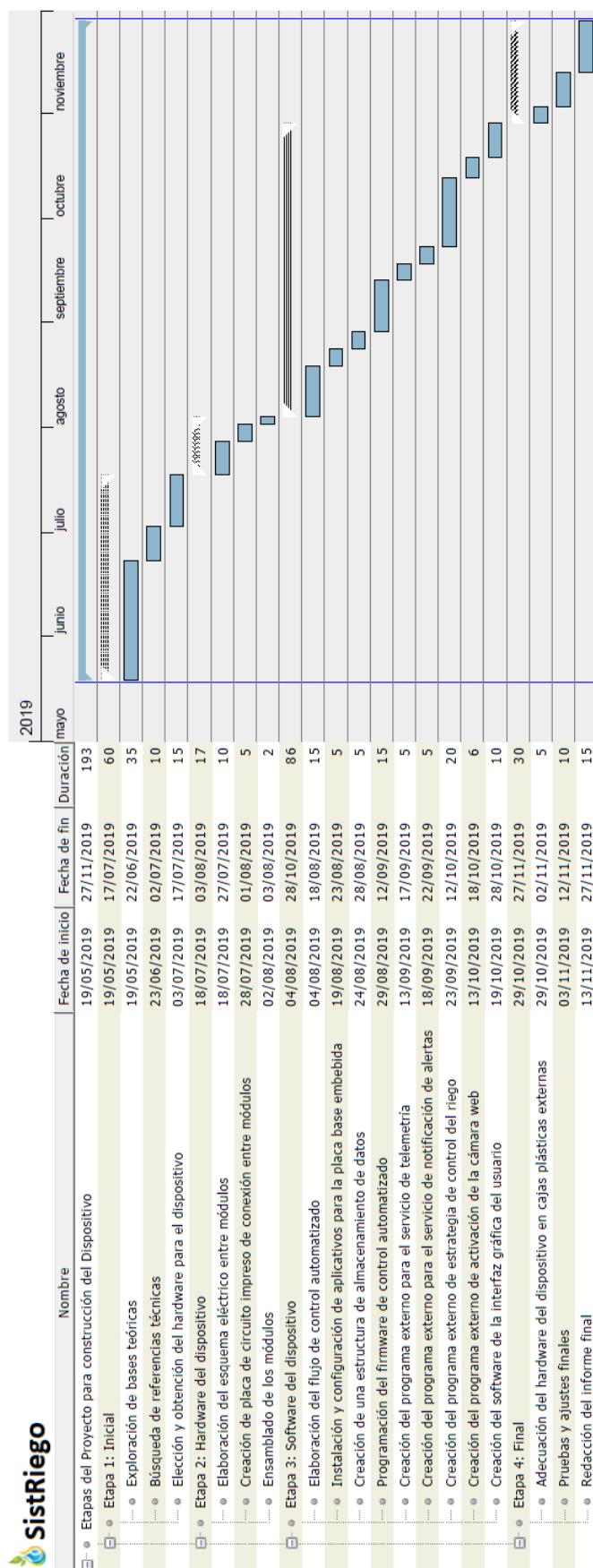
Etapa 4: Final.

Adecuación del hardware del dispositivo en cajas plásticas externas. Una vez finiquitado la etapa anterior se procederá a distribuir el hardware del dispositivo mediante cajas plásticas de conexión de uso externo agrupados según el esquema de un sistema de control automatizado: unidad de control, sensores, pre-actuador y actuador para protección y diferenciación operativa del dispositivo.

Pruebas y ajustes finales. Se procederá a verificar y corregir inconsistencias tanto en el hardware como en los softwares (firmware, servicios e interfaz del usuario) del dispositivo mediante pruebas de campo en un huerto urbano teniendo como base cinco macetas con cultivo de tomates, a fin de comprobar funcionamiento y comportamiento del mismo en la realidad intervenida.

Redacción del informe final. Como proceso final se redactará el informe final (tesis) mediante una herramienta informática de procesamiento de texto que permita la edición a través de LaTeX.

Diagrama de Gantt del desarrollo de las etapas



Plan de Trabajo

Desarrollo de las Etapas

En cada etapa del trabajo se describió lo que se hizo, cómo se hizo y que recursos fueron empleados: materiales, humanos y financieros, además al final del desarrollo de las etapas se presentan los tiempos de las etapas mediante un diagrama de gantt ejecutado y un resumen en cuanto a materiales, humanos y financieros por etapa mediante tablas.

Con respecto a los códigos fuentes de programas elaborados para el dispositivo SistRiego que tienen una considerable extensión para su entendimiento en forma impresa, decidí publicados en la plataforma en línea denominada GitHub que es un sistema de gestión de proyectos y control de versiones de código, así como una plataforma de red social diseñada para desarrolladores, por lo cual por cada programación hecha se mostrará un enlace para su visualización en forma adecuada e integra sin restricciones, además contiene todas las documentaciones propias del presente proyecto en la raíz del repositorio virtual.

Etapa 1: Inicial.

Exploración de bases teóricas Se realizaron análisis y estudios de bibliografías a fines al tema, apoyándome en libros y manuales (impresos y/o electrónicos), revistas, papers (artículos científicos), talleres, curricular de estudios universitarios mediante resúmenes técnicos realizados por docentes, como también de artículos electrónicos disponibles en internet mediante bibliotecas virtuales.

A continuación se muestra un resumen por cada tópico investigado.

- Ambiente físico de intervención

En relación al huerto urbano: Se obtuvo que al montar y mantener un huerto urbano en macetas en la propia casa otorga una cantidad de beneficios tanto para el individuo, su familia y a la comunidad misma, y sobre todo que los cultivos necesitan que se suministre el agua a intervalos regulares y en proporciones adecuadas a lo largo del día, afianzando así la necesidad de preservar este espacio verde mediante la automatización programable del riego según las condiciones ambientales.

En relación a las macetas: Se constató que el volumen de la maceta, sin importar el material con que esté fabricado, es importante para cada tipo de cultivo en relación directa a su forma radicular, además las necesidades de agua de cada especie varían, es decir, que la humedad del sustrato debe mantenerse dentro de los requerimientos de cada especie de cultivo, pero con respecto a la temperatura y la cantidad de luz solar que precisa cada especie de cultivo se encontró que existen especies como la variedad del tomate y del locote, entre otros, que soportan temperaturas e intensidades de luz solar alta y de especies que soportan temperaturas

bajas como el ajo, la cebolla, lechuga, zanahoria, entre otros, y que la única restricción es la de evitar el riego cuando la combinación de temperatura alta con una exposición alta de luz solar en forma directa supera a la permitida por la especie de cultivo, por más que la humedad del suelo lo necesite, debido al efecto lupa y refracción que genera el líquido con la luz solar pudiendo afectar al cultivo en su desarrollo.

En relación al riego por goteo mediante micro tubos: Se obtuvo que lo importante es inyectar una presión de agua con un bar de presión, la misma se obtiene mediante un reductor de presión, si la fuente de suministro supera dicha presión, y lo fundamental teniendo en cuenta este factor se precisan la cantidad de agua necesaria por especie de cultivo en la maceta por día en mililitros (ml), el caudal (volumen sobre tiempo) del gotero en mililitros por minuto (ml/min), la cantidad de veces que se hará el riego y por último el horario comprendido de riego en horas con minutos, es decir, desde que horario iniciará y terminará la jornada del riego en el día, con todos éstos datos se obtendrá el tiempo de riego activo en forma ideal, permitiendo personalizar el tiempo de espera entre riegos de acuerdo al horario.

Por tanto para un cultivo de tomate que se requiera 1.500 mililitros de agua al día por maceta repartidos como mínimo en un intervalo 3 veces al día, con un gotero con un caudal de 500 mililitros por minuto, se realiza primero la relación agua/veces resultando 1.500 mililitros dividido 3 veces al día se obtiene 500 mililitros por cada instancia de riego, con este resultado se efectúa una regla de tres simple, si el gotero en 1 minuto otorga 500 mililitros entonces para 500 mililitros de agua se debe activar el riego por el tiempo de 1 minuto tres veces al día para cumplir con la necesidad de 1.500 mililitros del agua día/maceta del tomate, el tiempo de espera se puede personalizar como mínimo 1 minuto entre riego y un máximo de los minutos restantes entre el horario de inicio y fin definido para el riego menos el tiempo total de activación, es decir, si se define que el riego se efectuará desde las 05:00 hs hasta las 06:00 hs se tiene 60 minutos totales, este valor se le resta los tiempos de activación que serían de 3 minutos arrojando 57 minutos restantes y por último éste resultado se divide por 3, la cantidad de veces en el día que se activará, nos da un tope de 19 minutos de espera entre riego, entonces el tiempo de espera que se puede estipular es de 1 a 19 minutos máximo.

Para calcular el caudal del gotero se debe tener un recipiente medidor en mililitros (0 a 500 ml o más) y colocar por la fuente de suministro de agua el reductor, por el reductor el micro-tubo, por el mircro-tubo el gotero y este último se introduce en el interior del recipiente a modo de que el mismo no toque el tope del agua en el recipiente al llenarlo, luego se procede a cronometrar hasta 1 minuto desde la apertura de la fuente de suministro la cantidad de agua vertida en el recipiente, luego de varios intentos se realiza la media y se tiene la cantidad de mililitros de agua que pasa por el gotero seleccionado en un minuto (caudal) según la fuente de suministro de agua.

Por último, de acuerdo a las consultas continuas en el tiempo de las condiciones ambientales fijadas mediante rangos mínimos y máximos permitidos por cada especie tanto en humedad del suelo, temperatura ambiente, iluminación diurna se deberá activar o desactivar el riego durante

el horario definido y en caso de lluvia desactivar sin importar los demás indicadores ambientales.

En caso de cancelación del riego antes o durante sin llegar a la cantidad de agua requerida por el cultivo en el día por las condiciones ambientales circundantes, la misma se deberá intentar las veces que sea necesario hasta cumplir con la meta del día en cantidad de agua, toda vez que no se utilice el límite de tiempo de espera, es decir, en el ejemplo anterior si se utiliza los 19 minutos solo podrá activar el riego 3 veces, es preferible crear varios intervalos de riego porque lo importante es suministrar el agua en forma gradual y puntual hasta la cantidad total de agua en el día.

- Sistemas de control y Automatización

En relación a sistemas de control: Se consideró para la construcción del dispositivo el sistema de control automático del tipo de lazo cerrado, utilizando el método avanzado en tiempo real y adaptativo, empleando la estrategia de control del tipo en cascada, con características de control experimental, clasificado como un sistema de control del carácter de transmisión en el tiempo.

En relación a la automatización: Se comprendió que el sistema de control automatizado consta de una unidad de control, sensores, pre-actuadores y actuadores, además de una interfaz para interactuar con el sistema. Por tanto el dispositivo creado utiliza lo expresado en su forma y funcionamiento integral.

- Fundamentos de la electricidad, electrónica e informática

Con relación a la electricidad: Se utilizaron los fundamentos de electricidad para el suministro de energía del dispositivo en relación al consumo de potencia y estabilidad.

Con respecto a la electrónica: Se emplearon los fundamentos de electrónica para comprender los componentes electrónicos básicos que integran un circuito electrónico en relación a comportamiento, medición, propiedades, aplicaciones y nomenclaturas que ayudaron a entender el funcionamiento de las partes electrónicas en sus procesos analógicos (voltaje y corriente) como también de los procesos digitales que utilizan; la unidad de control, los sensores, los pre-actuadores, los actuadores y demás partes electrónicas en relación a los pulsos de activación, control y medición. Por tanto, con todo lo mencionado se consiguió la correcta interpretación de las referencias técnicas de las partes electrónicas del dispositivo construido para su ensamblado, funcionamiento, prueba y ajustes de los mismos.

Con relación a la informática: Se utilizaron los conocimientos y experiencias adquiridas en la Universidad y en la profesión de Analista de Sistema Informáticos para enfocar la investigación del hardware, firmware y software open source para placas bases, sistemas operativos, lenguajes de programación, bases de datos e interconexión de redes para sistemas embebidos o empotrados, debido a que los sistemas de control de automatización utilizan este tipo de sistemas. En cuanto a la interfaz del usuario se decantó hacia la multiplataforma web mediante conexión inalámbrica mediante protocolo IEEE 802.11 para interactuar con el

dispositivo construido.

- Transferencia de video en tiempo real

Con respecto a la transferencia de video en tiempo real: Se obtuvo que la mejor manera de obtener video en tiempo real con mínimos requerimientos de recursos en un sistema embedido fue empleando una cámara web sobre el protocolo http, debido a que los sistemas operativos embebidos proporcionan controladores y aplicativos propios para la puesta en marcha en forma simple y robusta.

- Sistemas embebidos o empotrados

Con relación a sistemas embebidos o empotrados: Se obtuvo que el sistema operativo Linux en su distribución OpenWrt era la mejor opción para aplicar un sistema embebido o empotrado debido a que es un sistema operativo orientado a redes sin perder sus propiedades de personalización con respecto a: aplicativos, controladores y configuración, a más de soportar lenguajes de programación y herramientas informáticas robustas.

Búsqueda de referencias técnicas A partir del conocimiento adquirido con respecto a sistemas embebidos aplicados a sistemas de control de automatización se buscaron las partes para el dispositivo en tiendas virtuales en internet y en tiendas/comercios de electrónica locales, encontrando una amplia diversidad en la tecnología de placas bases embebidas para la unidad de control, por tanto se declinó hacia fabricantes reconocidos a nivel mundial, con una documentación y comunidad amplia como Arduino, Raspberry Pi y la Beagleboard.org, a partir de éstos me enfoqué en encontrar los módulos compatibles para las etapas de pre-actuadores, actuadores, sensores, componentes de tiempo real (reloj y cámara web), además de las fuentes de suministro de energía e indicadores luminosos necesarios para visualizar el estado del dispositivo, luego mediante sus características técnicas he agrupado cada una de las partes mediante tablas comparativas de por lo menos dos a tres modelos distintos para obtener un mejor criterio de elección a la hora de adquirir las piezas necesarias para el dispositivo y sobre todo orientados a las funcionalidades requeridas y deseadas.

Elección y obtención del hardware para el dispositivo. A continuación se explica el mecanismo de elección y obtención de las partes por separado:

El dispositivo construido consta de:

- Unidad de control: Se precisaba de una placa base de desarrollo que contenga un microcontrolador y un microprocesador independientes, con 9 pines digitales y 2 pines analógicos por lo menos, tener incorporado conexión inalámbrica IEEE 802.11, además de tener un sistema operativo Linux con distribución OpenWrt instalado o que pueda ser instalado mediante una tarjeta microSD, también de poseer dimensiones discretas, un puerto USB 2.0, consumo mínimo de energía eléctrica, software de desarrollo y subida de programas de control (firmware) desde de una computadora de escritorio con sistema operativo Windows 10. Se

optó por la placa base del fabricante Arduino en su modelo YÚN según características descritas en las tablas 12, 13, 14 y 15. La misma fue adquirida de la empresa MIC por Gs. 500.000.

- **Sensor de temperatura y humedad ambiente:** Se buscó un sensor que tenga mayor precisión de lectura y compatible con la familia Arduino, se optó por el fabricante Adafruit DHT22 según datos técnicos de la tabla 18. La misma fue adquirida de la empresa MIC por Gs. 50.000.

- **Sensor de humedad del suelo:** Se buscó un sensor que tenga menor consumo de energía, con un pin analógico de lectura y compatible con la familia Arduino, se optó por el fabricante SEN-HUS YL-69/YL-38 según datos técnicos de la tabla 19. La misma fue adquirida de la empresa MIC por Gs. 35.000.

- **Sensor de iluminación:** Se buscó un sensor que tenga el menor diámetro del tipo resistivo por su sencillez de montaje, se optó por el fabricante Nanyang Senba Optical Electronic CdS LDR GL5528 según datos técnicos de la tabla 21. La misma fue adquirida de la empresa MIC por Gs. 2.000.

- **Sensor de caudal (caudalímetro):** Se buscó un sensor que tenga el mismo diámetro de la electroválvula de 1/2 pulgada y compatible con la familia Arduino, se optó por el fabricante YIFA Sea YF-S201 según datos técnicos de la tabla 22. La misma fue adquirida de la empresa MIC por Gs. 80.000.

- **Sensor de detección de lluvia:** Se buscó un sensor que tenga un pin digital de lectura, con un área mayor de detección de gotas de lluvia y compatible con la familia Arduino, se optó por el fabricante SEN-HUS MH-RD/YL-38 según datos técnicos de la tabla 20. La misma fue adquirida de la empresa MIC por Gs. 35.000.

- **Pre-actuador:** Se buscó un pre-actuador de un solo canal de tipo electromecánico que maneje corriente continua (DC) y compatible con la familia Arduino, se optó por el fabricante Songle SRD-05VDC-SL-C según datos técnicos de la tabla 16. La misma fue adquirida de la empresa MIC por Gs. 40.000.

- **Actuador:** Se buscó un actuador del tipo electroválvula de corriente continua (DC) para operar el flujo de agua y por seguridad por no ser de corriente alterna (AC), además de tener un costo menor, se optó por un modelo genérico según datos técnicos de la tabla 17. La misma fue adquirida de la empresa MIC por Gs. 60.000.

- **Reloj de tiempo real:** Se buscó un módulo de tiempo real RTC debido a que el sistema

de control se basa en el control en el tiempo transcurrido (fecha - hora), y la placa base de desarrollo no incluyen módulo de retención de fecha hora, es decir, una vez apagada las fecha hora del sistema vuelve a la fecha hora de fabricación, por tal motivo se buscó con una precisión respetable pero de costo menor y compatible con la familia Arduino, se optó por el fabricante Electrow RTC DS1307 según datos técnicos de la tabla 22. La misma fue adquirida de la empresa MIC por Gs. 35.000.

- **Indicadores luminosos:** Se buscaron indicadores luminosos simples para permitir al usuario detectar visualmente los estados del dispositivo tanto de la unidad de control, del pre-actuador y del actuador, se optó por el tipo LED-01 de color Rojo y Verde dos de cada uno, además de un LED RGB-B5C según datos técnicos de la tabla 23. La misma fue adquirida de la empresa MIC en total por Gs. 3.000.

- **Tarjetas de almacenamiento:** Se buscó una tarjeta de almacenamiento del tipo MicroSD de 16 GB de óptimo rendimiento, se optó por el fabricante Sandisk SDSQAF3-008G-I según datos técnicos de la tabla 24. La misma fue adquirida de la empresa MIC por Gs. 60.000.

- **Cámara web:** Se buscó un periférico de entrada de video captura web de calidad y con un costo accesible, se optó por el fabricante Logitech C270 según datos técnicos de la tabla 25. La misma fue adquirida de la empresa Multimedia por Gs. 150.000.

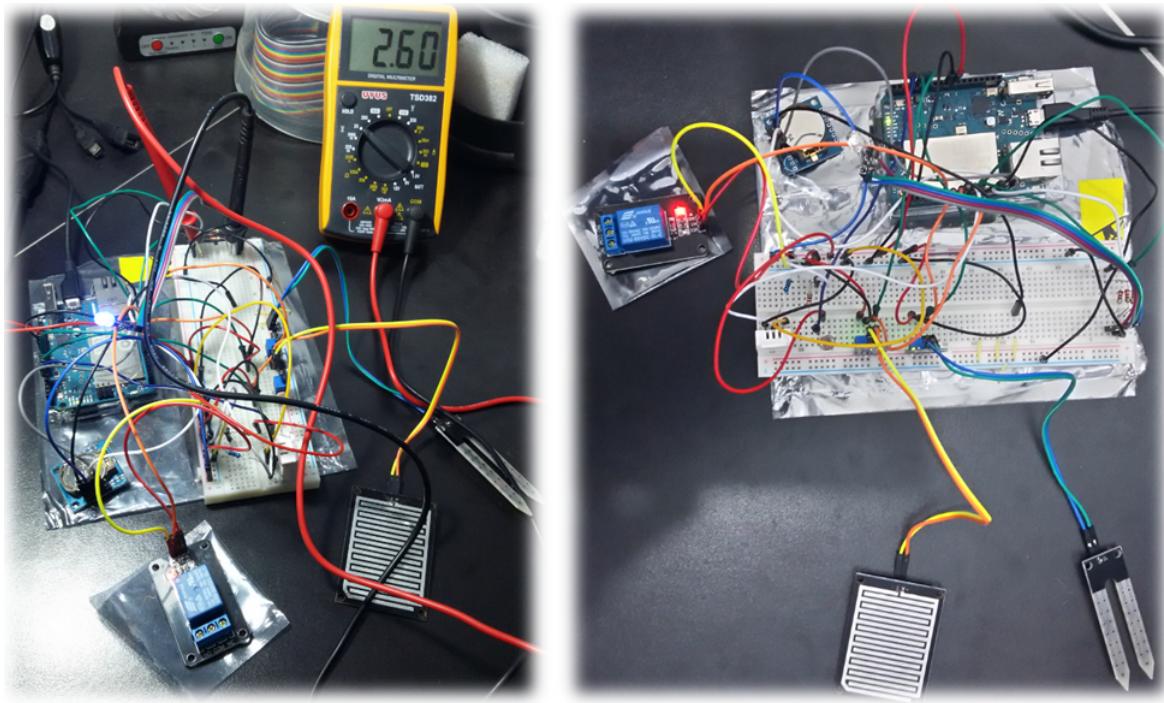
- **Fuentes de suministro de energía:** Se buscó dos fuentes de alimentación con suministro de energía estable y confiable, se optó por el modelo UBP-008 según datos técnicos de la tabla 26. La misma fue adquirida de la empresa MIC por Gs. 35.000 cada uno.

Etapa 2: Hardware del dispositivo.

Elaboración del esquema eléctrico entre módulos. Una vez adquirido todos los módulos necesarios para el dispositivo, se procedió a desempaquetar de sus cajas/sobres, observar cada módulo con sus indicaciones técnicas, las mismas se encuentran resumidas en el Anexo C, por último se procedió a montar las conexiones entre los módulos según pines del Arduino Yún con los datos proporcionados por los fabricante de los demás módulos adquiridos, para ésta tarea se utilizó un tablero de pruebas (Protoboard) y cables del tipo dupont macho-macho para realizar las conexiones entre la unidad de control y sus módulos (reloj de tiempo real, sensores, pre-actuador e indicadores luminosos), para ordenar y clarificar las conexiones de pines se creó una tabla de disposición de interconexión entre la unidad de control y sus módulos, como se aprecia en la tabla de abajo.

Tabla 29: Disposición de conexión de pines del Arduino Yún. Fuente: Autor

Figura 42: Montaje de los módulos del dispositivo en protoboard. Fuente: Autor



Una vez terminado la interconexión con los módulos al Arduino Yún según sugerencias técnicas de los fabricantes y como se aprecia en la figura de arriba se procedió a la creación del esquema eléctrico mediante una herramienta web en línea de suscripción gratuita denominada EasyEDA (<https://easyeda.com/>), la misma se enfoca en el proyecto, concepción, y producción de sistemas electrónicos, abarcando desde el proyecto de circuitos integrados hasta el desarrollo de placas de circuito impreso, con la mencionada herramienta se creó el diagrama de esquema eléctrico de interconexión de las partes electrónicas del dispositivo como se aprecia en la figura de abajo.

Figura 43: Diagrama electrónico o esquema eléctrico del dispositivo. Fuente: Autor

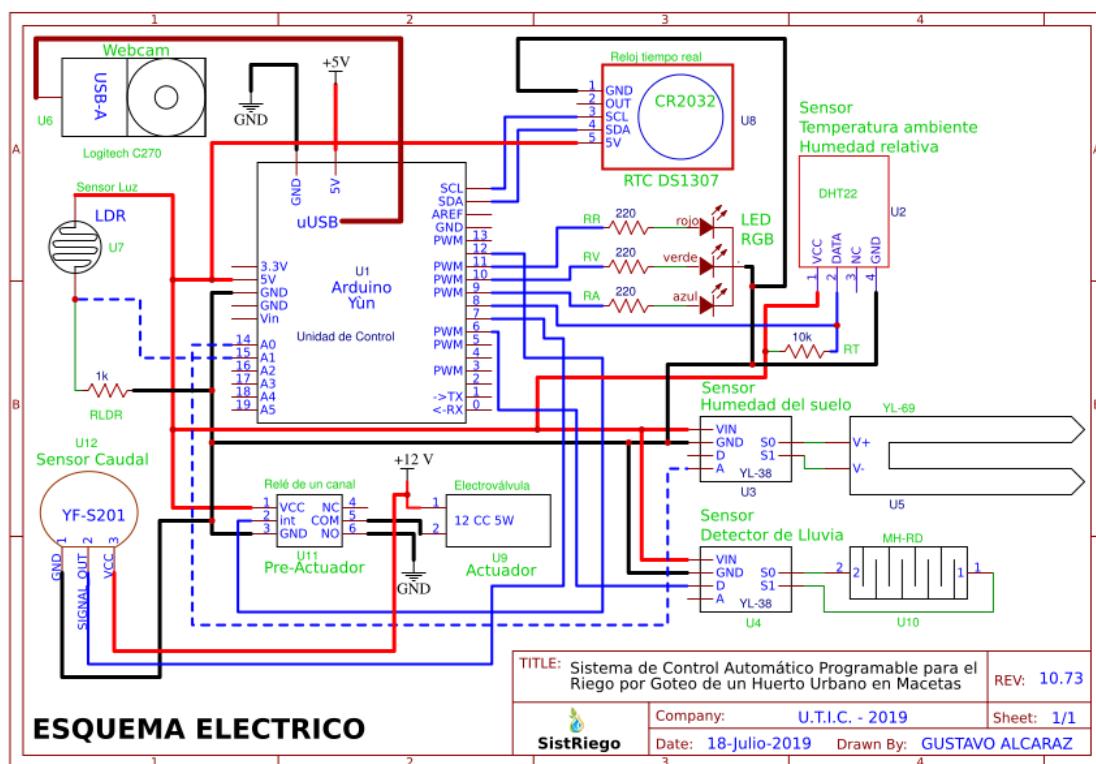
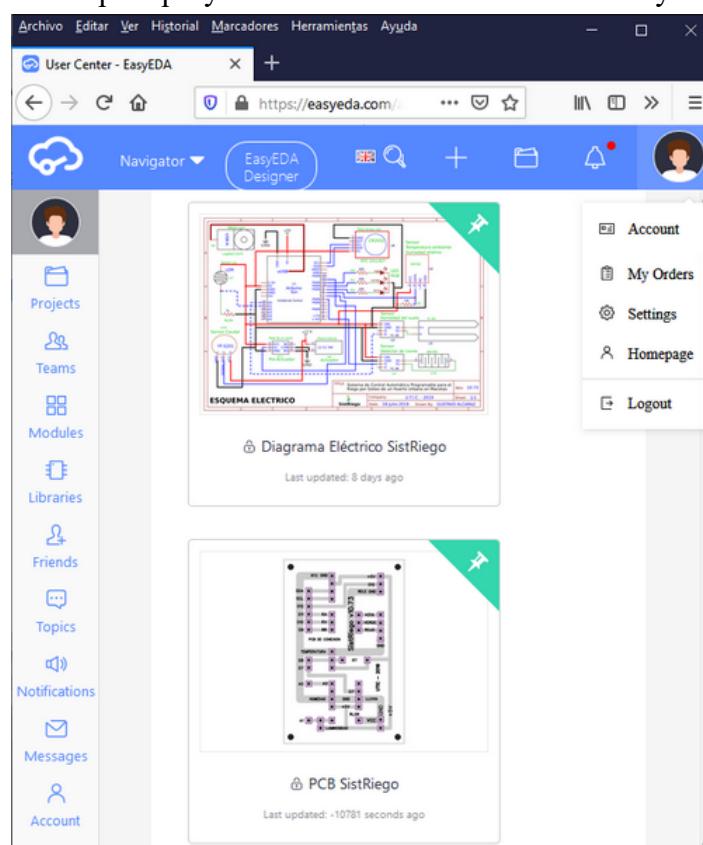


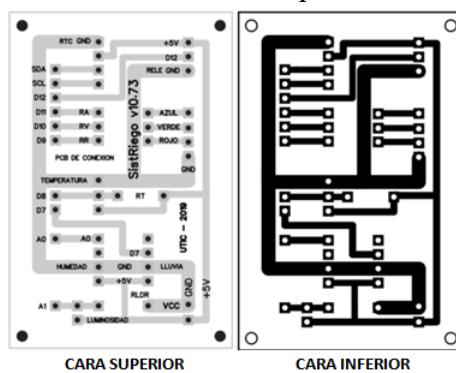
Figura 44: Herramienta para proyectos de sistemas electrónicos EasyEDA. Fuente: Autor.



Creación de placa de circuito impreso de conexión entre módulos. A continuación se explican los pasos para la creación de una placa de circuito impreso (PCB):

Diseño de pistas para el circuito impreso: con el esquema eléctrico concebido se procedió a la creación de las pistas de conexión de la placa de circuito impreso (PCB), utilizando la misma herramienta EasyEDA para diseñar en forma equitativa, armónica, discreta y compacta los conectores y componentes necesarios como se aprecia en la figura de abajo, teniendo una dimensión final de 4,0 x 5,5 cm.

Figura 45: Pistas del PCB del dispositivo. Fuente: Autor



Transferencia del diseño a una placa virgen de mica: se procedió a imprimir en un papel fotográfico de tipo revista la cara inferior del diseño mediante el uso de una impresora láser, con las dimensiones del diseño impreso se recortó una placa virgen de mica, posteriormente se colocó la parte impresa sobre la superficie de cobre de la placa de mica quedando superpuestas, por medio de la técnica del planchado, es decir, empleando una plancha eléctrica a temperatura alta, se planchó uniformemente empezando por las esquinas del papel superpuesto hasta pasados unos 5 minutos produciéndose así el efecto de termo transferencia de la pista del diseño impreso a la superficie de cobre de placa de mica, una vez terminado la técnica, el papel queda adherido a la placa de mica, para remover el papel se sumergió la misma en un recipiente con agua, se esperó alrededor de 10 minutos y luego con la ayuda de las yemas de los dedos de la mano se friccionó hasta retirar todo el papel de la placa dentro del recipiente, después se sacó del recipiente y se secó con papel de cocina, paso seguido se verificó que las pistas de diseño trasferidas estén bien definidas y marcadas, se corrigió algunos defectos en la misma pasando sobre las imperfecciones un marcador permanente negro hasta cubrir bien los defectos.

Eliminación química del cobre no cubierto de la placa de mica: en este paso se necesitó diluir medio kilo en polvo de cloruro de hierro en un recipiente de vidrio, en mi caso utilicé un recipiente de café instantáneo de 500 gramos por que el tamaño del PCB permite su acceso al recipiente y una vez utilizado se tapa para mantener sus propiedades por mucho tiempo resguardado en un lugar seco y oscuro, con medio litro (500 ml) de agua, en este proceso se tuvo mucho cuidado en la manipulación tanto de no aspirar los gases que emana ni tocar el líquido resultante (ácido percloruro férrico) y sobre todo se manipuló en un ambiente ventilado

y abierto, luego se perforó con un taladro la placa de mica en una esquina con una mecha para metal de 1 milímetro (mm) y se introdujo por el orificio un hilo de nailon (hilo de pescar) de unos 30 cm, luego sosteniendo del hilo por el extremo se sumergió la placa de mica en el interior del recipiente, cada 5 minutos se verificó el estado de la placa retirando mediante el hilo, esto se repitió hasta que quedó solamente las pistas marcadas, confirmado la eliminación química del cobre no cubierto se procedió a limpiar bajo el fregadero con agua sin tocar con las manos solo por el hilo de nailon, una vez enjaguado se procede a secar con papel de cocina, por último se removió las pistas cubiertas frotando con una virulana hasta visualizar el cobre de la placa de mica.

Orificios para conectores y componentes del PCB: en este último paso se procedió a taladrar con una mecha para metal de 1 mm los orificios de componentes/conectores necesarios y con una mecha para metal de 3 mm los agujeros de fijación de las esquinas según el diseño impreso en la placa de mica. Abajo en la figura se aprecian los trabajos realizados para el PCB.

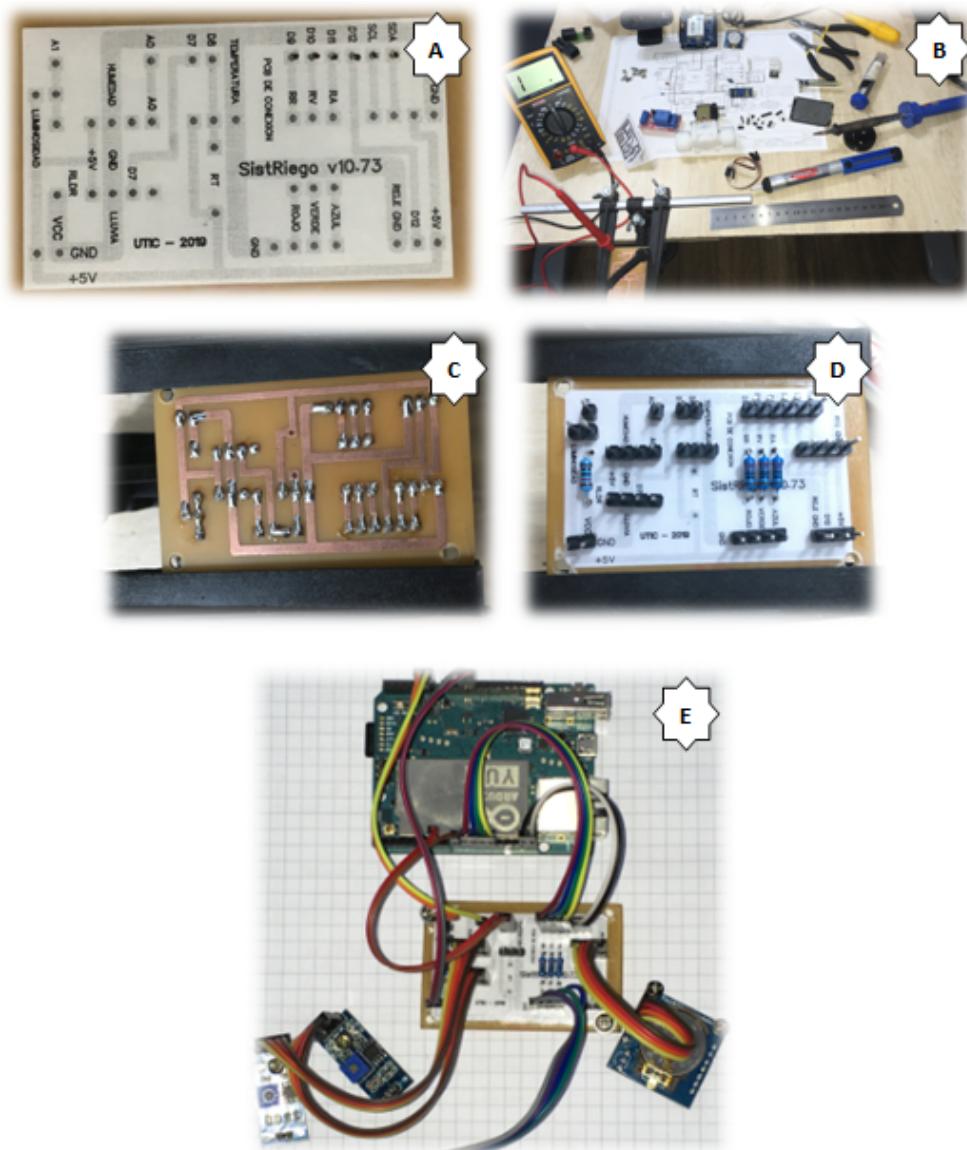
Figura 46: Proceso del PCB para el dispositivo. Fuente: Autor



Ensamblado de los módulos. Con la creación de la placa de circuito impreso (PCB) para las conexiones entre la unidad de control y los demás módulos se procedió a imprimir el diseño de la cara superior, luego de recortado se adhirió en la cara opuesta a las pistas del PCB

mediante papel contact transparente, después se perforó el plastificado con una aguja en los orificios correspondientes, posteriormente se soldó los conectores y componentes por la placa de circuito impreso utilizando un soldador de 30 watt con estaño de carrete azul (aleación de estaño 60% con plomo 40%), además se empleó un soporte para placa y alicates, terminado este proceso se realizaron las conexiones entre la unidad de control (Arduino Yún) y los módulos del dispositivo, por último se verificaron los voltajes y la conductividad de las pistas entre los pines de la unidad de control y sus módulos mediante un multímetro. En la figura de abajo se aprecia el proceso de ensamblado final. En la etapa final se realizó el montaje en cajas plásticas externas con el fin de resguardar el hardware para las pruebas y ajustes finales.

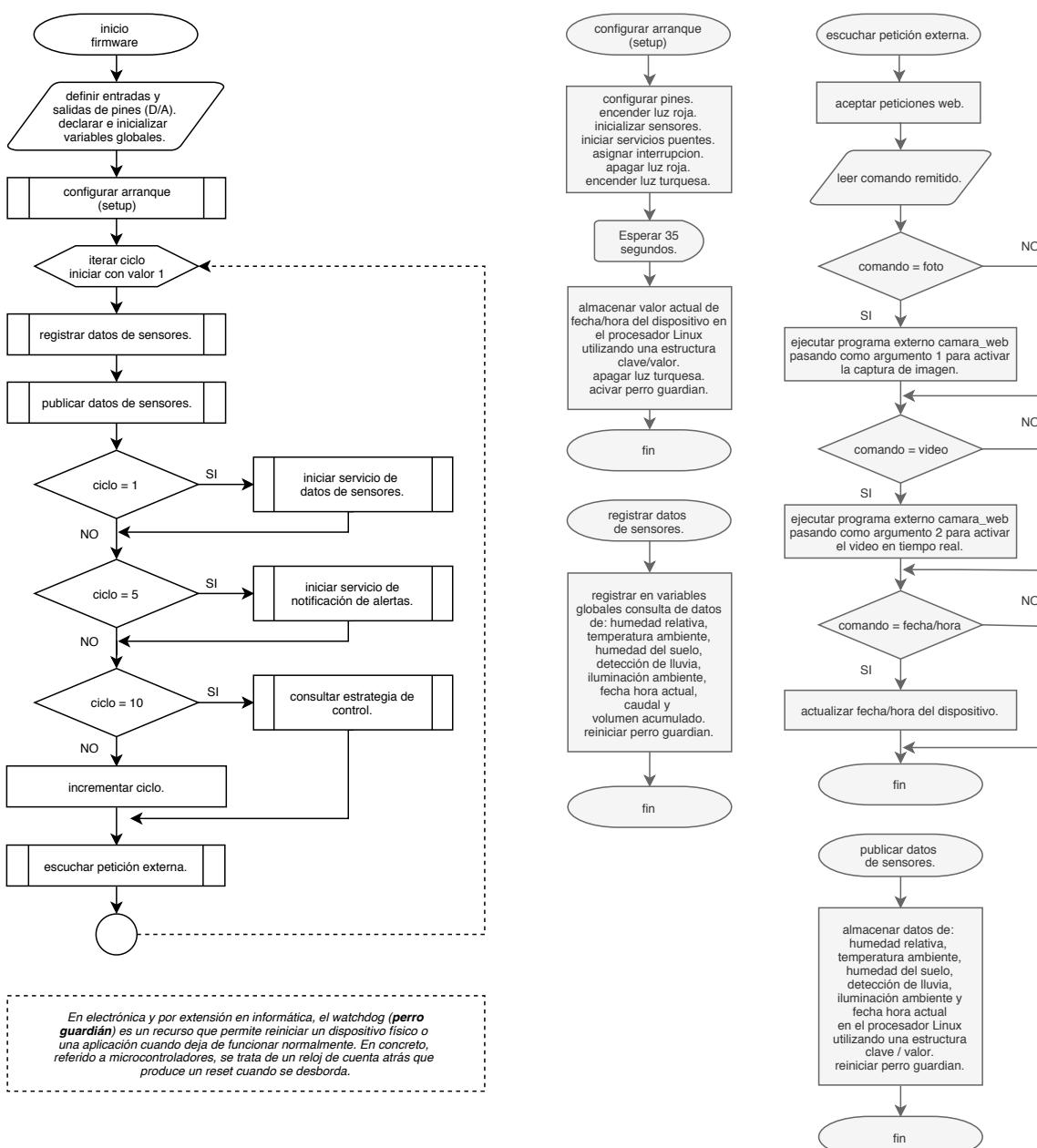
Figura 47: Proceso del ensamblado de los módulos. Fuente: Autor



Etapa 3: Software del dispositivo.

Elaboración del flujo de control automatizado. El flujo de control automatizado se representó gráficamente mediante un diagrama de flujo principal que corresponde al Firmware del dispositivo con sus respectivos diagramas de flujos alternos y por cuatro diagramas de flujo que corresponden a los programas externos que se ejecutan en el Firmware, los diagramas fueron creados mediante la herramienta web de código abierto draw.io, con enlace url <https://www.draw.io/>, que permitió crear aplicaciones de diagramación en forma sencilla y robusta, éstos diagramas de flujo fueron elaborados a partir de los conocimientos adquiridos en la teoría y gran parte por la creatividad del autor para adaptar la electrónica con la informática.

Figura 48: Diagrama de flujo principal del Firmware para el dispositivo. Fuente: Autor.



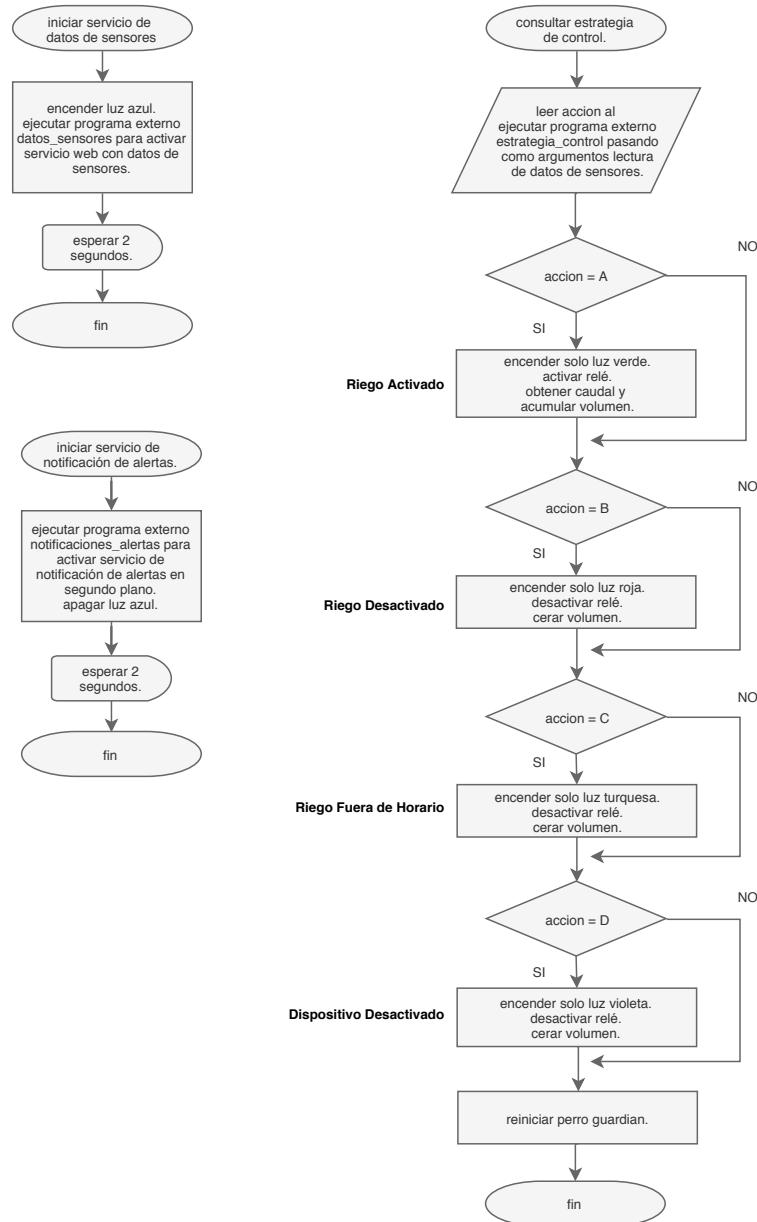


Figura 49: Diagrama de flujo del programa externo para servicio de datos de sensores (datos_sensores). Fuente: Autor.

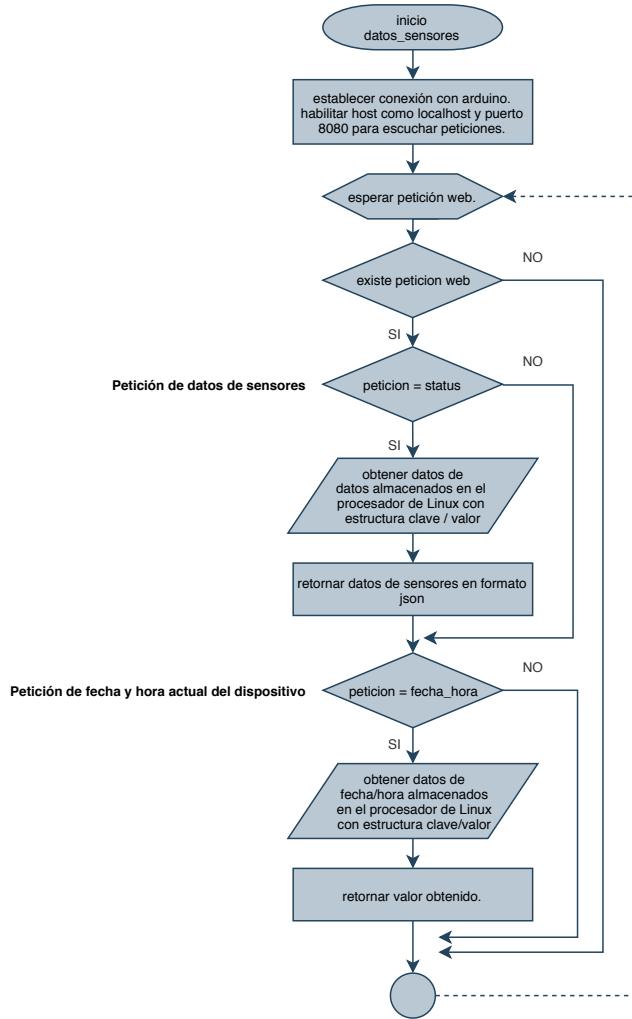


Figura 50: Diagrama de flujo del programa externo para el servicio de notificación de alertas (notificaciones_alertas). Fuente: Autor.

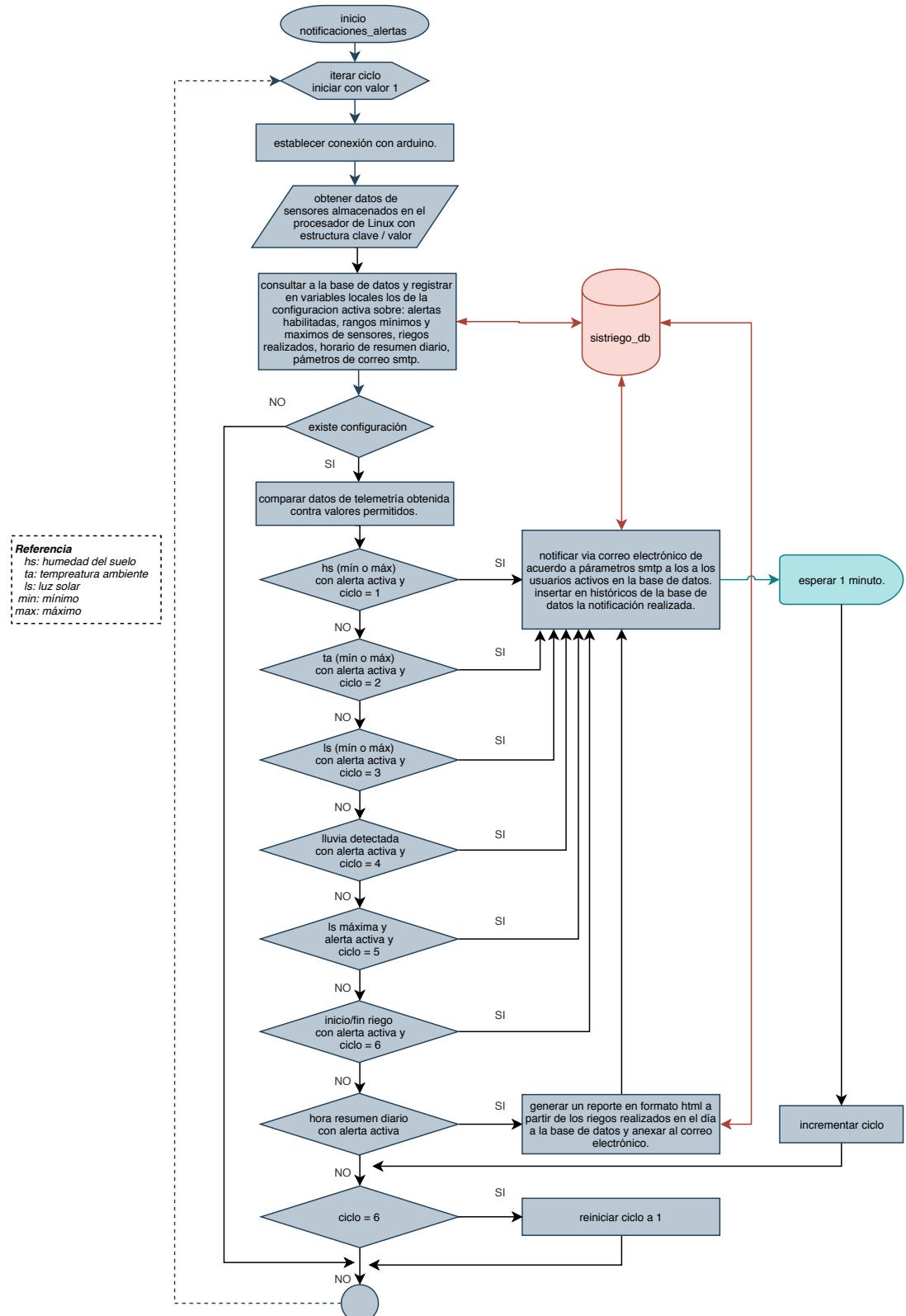


Figura 51: Diagrama de flujo del programa externo para la estrategia de control del riego (estrategia_control). Fuente: Autor.

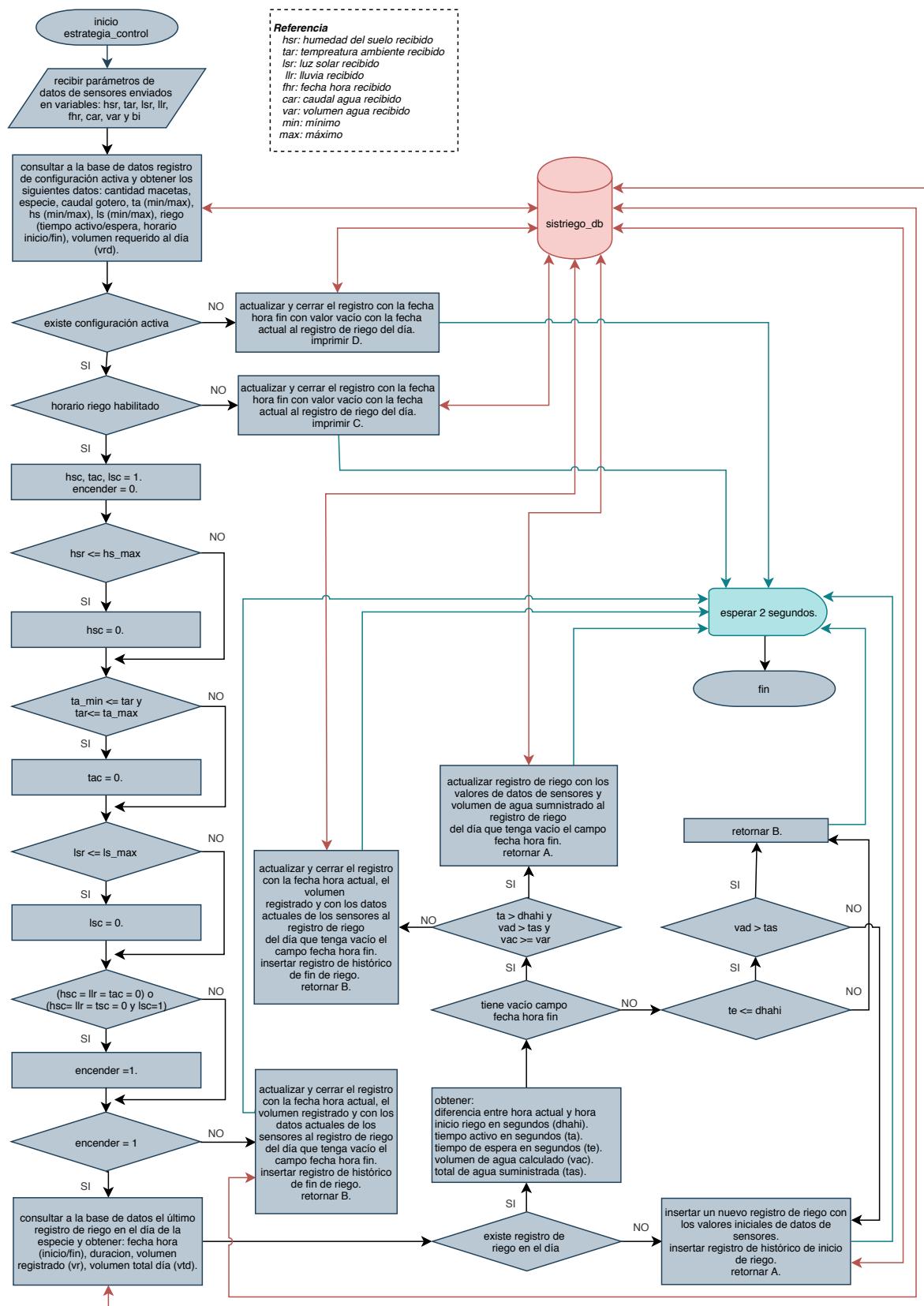


Figura 52: Diagrama de flujo del programa externo para utilizar la cámara web (camara_web).
Fuente: Autor.

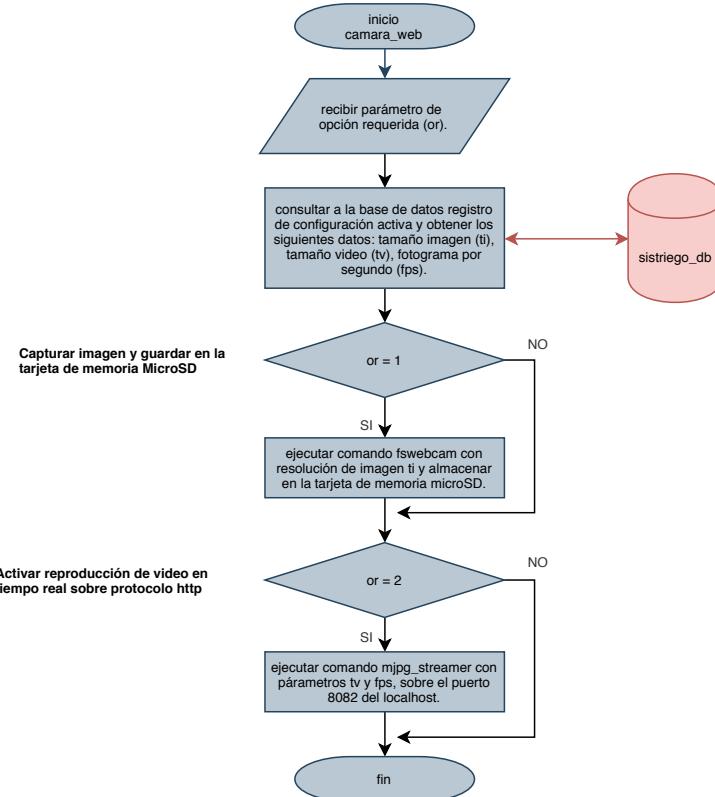
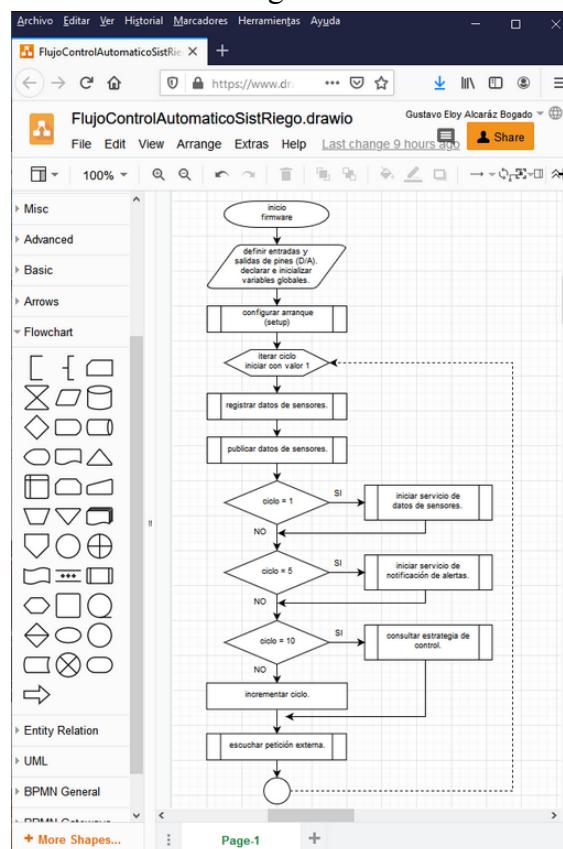


Figura 53: Herramienta de diagramación draw.io. Fuente: Autor.

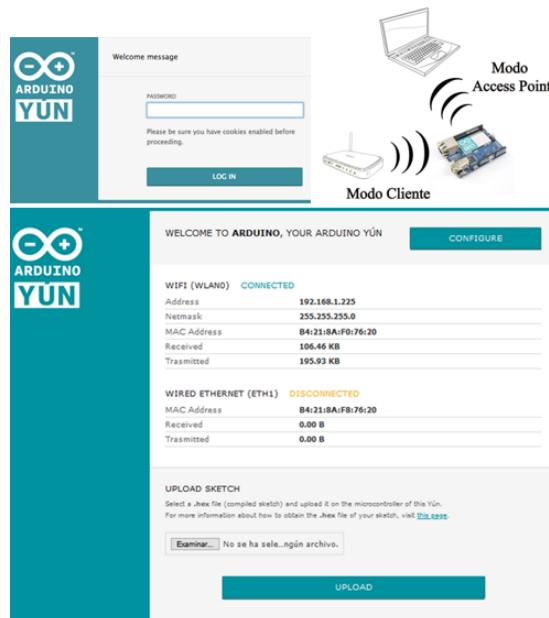


Instalación y configuración de aplicativos para la placa base embebida. En este tópico se instalaron y configuraron los aplicativos necesarios para comunicar, preparar y utilizar la placa base embebida del Arduino Yún de acuerdo a los requerimientos del dispositivo propuesto.

En primer lugar se procedió a energizar la placa base de Arduino Yún para su primera configuración, conectándolo al puerto USB de mi computadora de escritorio mediante el cable USB suministrado por el fabricante, una vez habilitado la red inalámbrica de la placa (el led azul de la placa se activa), se observó en las redes inalámbricas de Windows una red Wifi disponible llamada “Arduino Yun-B4218AF07620”, se conectó a dicha red y se procedió a lanzar el navegador Firefox instalado en mi computadora de escritorio, en la barra de URL o de direcciones se colocó <http://192.168.240.1> y se pulsó la tecla enter, una vez que se accedió a la consola apareció un recuadro de contraseña, en el cual se tecleó la clave “arduino” de acuerdo al fabricante, luego de aceptado la contraseña se visualizó una ventana desde la cual fue posible configurar el Arduino Yún para establecer un nombre y una contraseña nueva para el acceso a la placa, en mi caso decidí mantener el nombre y agregarle la palabra “arduinoyun” a la clave nueva, posteriormente se estableció una red de Internet vía WiFi provista en mi casa desde la cual el Arduino Yún se pudo conectar a la web, una vez que se estableció la red y la contraseña se finalizó la configuración, terminado ésto emergió una ventana que indicaba que se debía conectar a la misma red WiFi que se estableció en el Yún y con ésto quedó configurado y listo para ser utilizado la placa base embebida de Arduino Yún.

De ésta forma se demuestra que el Arduino Yún puede comunicarse vía red WiFi tanto como modo cliente (comunicación indirecta) o como modo de punto de acceso inalámbrico (comunicación directa) y acceder al mismo mediante un navegador web.

Figura 54: Configuración del Arduino Yún. Fuente: Autor



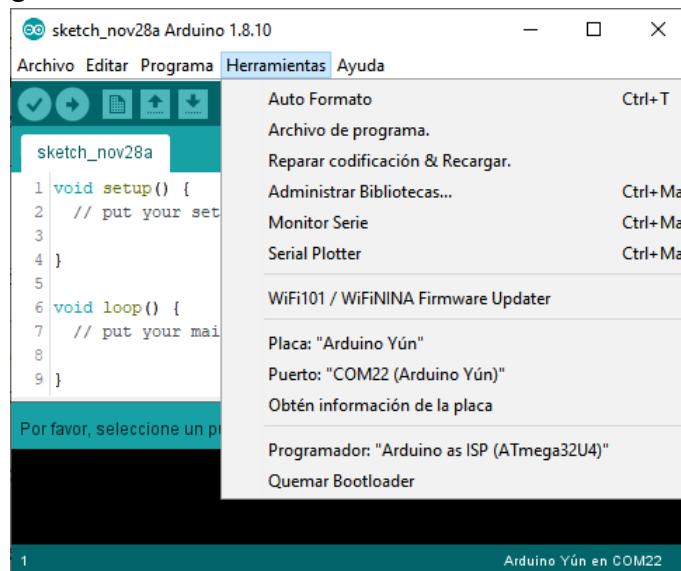
A continuación se explica las herramientas instaladas y utilizadas en mi computadora de escritorio con sistema operativo Windows 10 de 64bits:

- **IDE Arduino v1.8.10:** el IDE de desarrollo de Arduino emplea la plataforma Wiring/Processing que consta de un nombre de fichero con una extensión “.ino” al cual se denomina sketch (boceto) que tiene una estructura comprendida por una función setup() y otra loop(), además utiliza un compilador GCC para C/C++, es decir, se programa el sketch en lenguaje de programación en C/C++ simplificado para facilitar las operaciones de entrada y salida debido a que es más sencillo la comunicación entre el plano de la física y la electrónica con el dispositivo mediante librerías específicas según el módulo electrónico adquirido. En tanto la función setup() se emplea para la configuración de los módulos y la misma se invoca solo al iniciar la placa Arduino, mientras que la función loop() se colocan las sentencias que repetirán como bucle infinito en el dispositivo o mejor dicho hasta que vuelva a reiniciar la placa Arduino.

El IDE de desarrollo Arduino está escrito en JAVA para lo cual se descargó e instaló una versión para el sistema operativo Windows 10 de 64bits de la url <https://www.arduino.cc/en/Main/Software>, una vez instalado se procedió a configurar la placa Arduino Yún conectando la misma a mi computadora por el puerto USB, una vez detectado por la computadora se procedió a ejecutar la aplicación Arduino, en la aplicación se seleccionó en el menú Herramientas la placa Arduino Yún y el puerto COM22 en mi caso, este modelo en particular tiene la posibilidad de conectarse tanto por cable USB como por WiFi, pero por seguridad recomiendo por cable, debido a que por fallas de cualquier tipo en el enlace se puede bloquear la placa al momento de subir el sketch en forma parcial por interrupción antes de terminar el proceso, además se debe contar con una UPS para evitar este mismo efecto por corte o pestaño de energía eléctrica.

En el proceso de subida del sketch se realiza la compilación del fichero a un código hexadecimal y una vez terminado, toda vez que no sea detectado ningún error en el código programado, lo envía a la placa Arduino Yún en donde la misma la transforma en código de máquina (binario) para su grabado dentro de la memoria flash del microcontrolador para su funcionamiento, una vez configurado se pudo crear el sketch del firmware y subirlo a la placa Arduino Yún sin ningún inconveniente, las veces que fuere necesario.

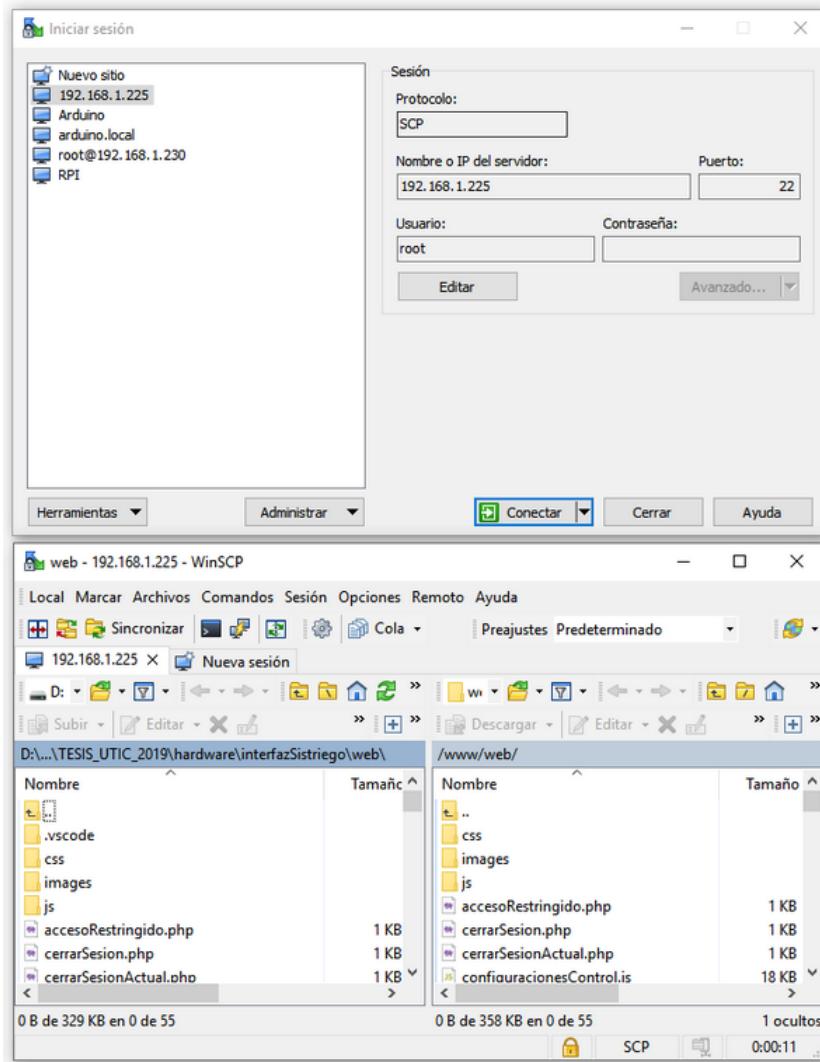
Figura 55: IDE de desarrollo Arduino Yún. Fuente: Autor



- **WinSCP v5.15.7:** WinSCP es una aplicación libre y de código abierto, consistente en un cliente SFTP gráfico para Windows que emplea SSH, también soporta el anterior protocolo SCP. Los pasos a seguir para subir archivos a un servidor por SSH utilizando WinSCP: colocar la IP o el nombre del servidor: en mi caso 192.168.1.225 (IP de la placa Arduino Yún en mi red particular) Usuario: root. Contraseña: arduinoyun. “Protocolo”->“Archivos de Protocolo”: aquí se cambió por SCP, al conectarse aparecen dos ventanas, una muestra la vista de ficheros locales y la otra los ficheros del servidor remoto conectado y a partir de ahí se buscan los ficheros a ser transferidos en las ubicaciones de carpetas deseadas.

Ésta herramienta fue descargada e instalada desde la url <https://winscp.net/eng/download.php>, con la misma fue posible levantar todos los ficheros necesarios como la base de datos SQLite, los programas externos hechos en el lenguaje de programación Python y toda la estructura web hecha en PHP para la interfaz gráfica del usuario, además de comprender mejor la organización y distribución de los ficheros del S.O. Linux OpenWrt.

Figura 56: WinSCP. Fuente: Autor

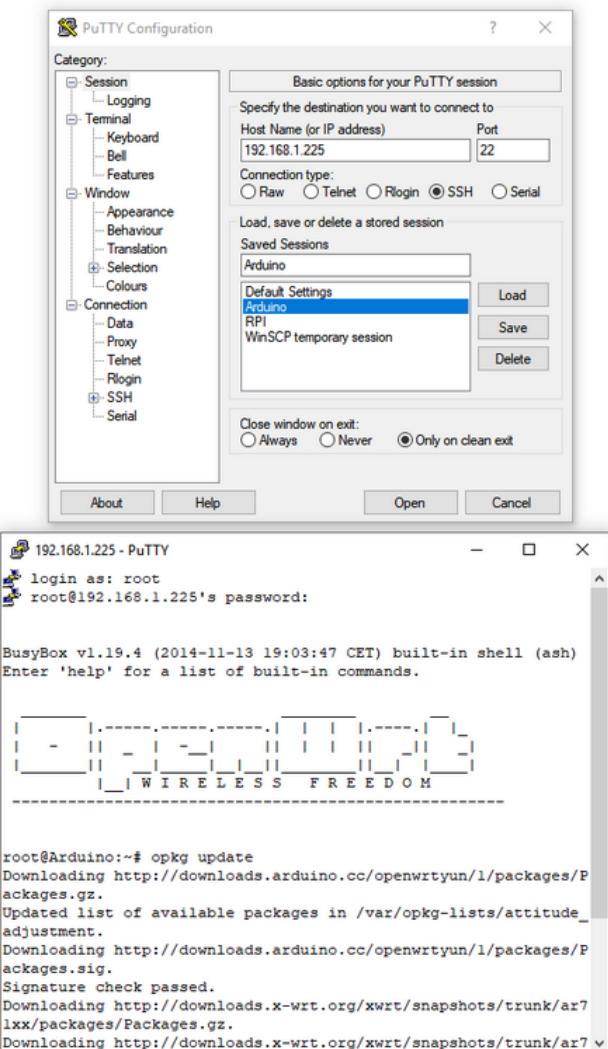


- **PuTTY v0.73:** PuTTY es un cliente SSH y Telnet con el se puede conectar a servidores remotos iniciando una sesión en ellos permitiendo ejecutar comandos. Se ha empleado PuTTY para ejecutar comandos como terminal local en el S.O. Linux OpenWrt y así poder instalar los paquetes listados en la Tabla 30, necesarios para el funcionamiento deseado en el dispositivo propuesto, mediante la instrucción “opkg install nombre de paquete”, opkg es el sistema de gestión de paquetes ligeros basado en ipkg del Linux OpenWrt que cuenta con más de 3.000 paquetes disponibles, previamente se debe actualizar el repositorio de paquetes mediante la instrucción “opkg update” antes de cualquier instalación de paquetes. Ésta herramienta fue descargada e instalada desde la url <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>, por último esta herramienta es un complemento compatible con WinSCP.

Tabla 30: Lista de paquetes instalados en el S. O. Linux OpenWrt del Arduino Yún mediante el aplicativo PuTTY. Fuente: Autor

Programa	Paquete	Versión	Observación
SQLITE Sistema de gestión de bases de datos relacional utilizado para almacenar la estructura de datos del dispositivo.	libsdlite3	3071201-1	Por último se ejecuta en la consola ssh: mkdir /mnt/sda1/SistRiego mkdir /mnt/sda1/SistRiego/sqlite Y por WinSCP se transfiere a esta carpeta la base de datos creada en SQLite
	sqlite3-cli	3071201-1	Para ejecutar el programa desde la consola ssh de debe escribir sqlite3 mnt/sda1/SistRiego/sqlite/sistriego_db.db
PYTHON Lenguaje de programación multiparadigma utilizado para crear los programas externos que se ejecutan en el firmware del dispositivo.	python	2.7.3-2	Se descargaron desde Internet y se agregaron a la carpeta con ruta remota /usr/lib/python2.7/ las siguientes librerías:
	python-json	3_4-1	bottle (https://bottlepy.org), request (git://github.com/kennethreitz/requests.git),
	python-mini	2.7.3-2	idna (https://pypi.org/project/idna/), chardet (https://pypi.org/project/chardet/),
	python-openssl	2.7.3-2	certifi (https://pypi.org/project/certifi/) y urlib3 (https://pypi.org/project/urllib3/)
	python-sqlite3	2.7.3-2	Por último se ejecuta en la consola ssh: mkdir /mnt/sda1/SistRiego/python
PHP Lenguaje de programación interpretado del lado del servidor utilizado para crear la interfaz gráfica del usuario del dispositivo mediante páginas web.	php5	5.4.5-3	Una vez terminado las instaciones de paquetes para PHP se ejecutaron las siguientes líneas de comandos:
	php5-cgi	5.4.5-3	uci set uhttpd.main.interpreter=".php=/usr/bin/php-cgi" uci set uhttpd.main.index_page="index.html index.htm default.html default.htm index.php"
	php5-cli	5.4.5-3	uci commit uhttpd
	php5-mod-curl	5.4.5-3	sed -i 's,doc_root.*,doc_root = "",g' /etc/php.ini
	php5-mod-json	5.4.5-3	sed -i 's,;short_open_tag = Off,short_open_tag = On,g' /etc/php.ini
	php5-mod-session	5.4.5-3	/etc/init.d/uhttpd restart
	php5-mod-sqlite3	5.4.5-3	mkdir /mnt/sda1/SistRiego/web ln -s /mnt/sda1/SistRiego/web/ /www/ In -s /mnt/sda1/SistRiego/web/ /www/
USB Unidad de expansión para el núcleo Linux requerido para detectar y reconocer dispositivos USB 2.0	kmod-usb-core	3.3.8-1	En este caso se instaló para detectar y reconocer la cámara web Logitech c270 en el puerto USB de la Placa Arduino Yún
	kmod-usb-storage	3.3.8-1	
	kmod-usb2	3.3.8-1	
	libusb	0.1.12-3	
	libusb-1.0	1.0.9-1	
VIDEO Unidad de expansión para el núcleo Linux requerido para captura de imagen y manejo de video	usbutils	005-1	Ejemplo de uso del capturador de imagen por webcam: fswebcam -r "800x600" /mnt/sda1/SistRiego/web/imagen.png Ejemplo de uso de la webcam como video en tiempo real: mjpg_streamer -i input_uvc.so -d /dev/video0 -r "800x600" -f "15"-o output_http.so -p 8082 -w /www/webcam
	kmod-video-core	3.3.8-1	
	kmod-video-uvc	3.3.8-1	
	kmod-video-videobuf2	3.3.8-1	
	fswebcam	20110717-1	
	mjpg-streamer	r148-4	

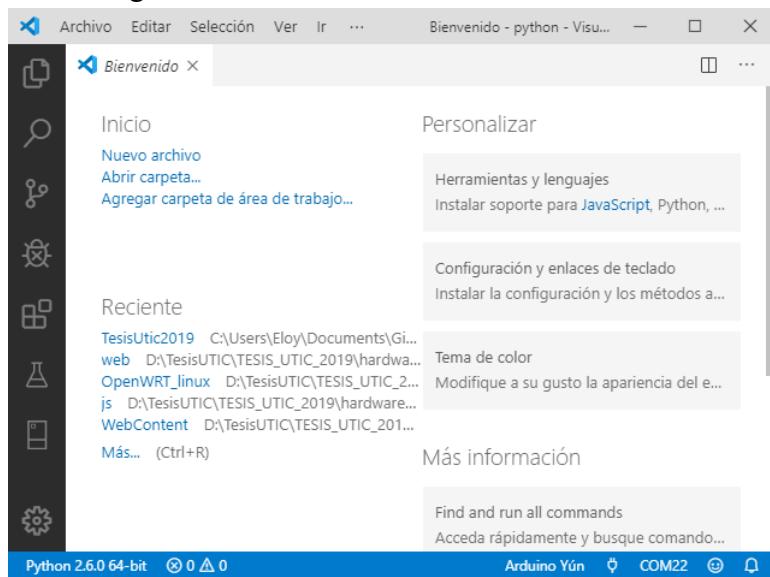
Figura 57: PuTTY. Fuente: Autor



- Visual Studio Code v1.40.2: Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows , Linux y macOS, incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. Ésta herramienta fue descargada e instalada desde la url <https://code.visualstudio.com/Download>.

Ésta herramienta fue utilizada para crear los programas externos escritos con sintaxis del lenguaje de programación Python que son ejecutados por el firmware del dispositivo construido.

Figura 58: Visual Studio Code. Fuente: Autor



Preparación de una estructura de almacenamiento de datos. Una vez obtenido los diagramas de flujos de control automatizado se constató la necesidad de tener una estructura de almacenamiento de datos para el dispositivo (Ver Figuras 49, 50, 51 y 52), por lo cual se diseñó un diagrama de entidad relación simple que consta de los siguientes nombres de entidades con sus atributos, para crear el diagrama de entidad relación se utilizó la herramienta draw.io:

especies: para almacenar los datos de las especies de cultivo como: nombre, descripción (nombre científico), los rangos mínimos y máximos permitidos por la misma con respecto a temperatura ambiente ($^{\circ}\text{C}$), humedad del suelo (%) y luz solar (%), la cantidad de agua requerida por maceta y la frecuencia de riego (veces).

configuraciones: para almacenar las configuraciones propias del dispositivo en cuanto al tipo de especie de cultivo, el tipo y la dimensión de la maceta, la cantidad de macetas, el tiempo de riego activo y de espera entre riego, el caudal del gotero utilizado, el horario de riego (inicio/fin), el horario de envío del correo electrónico para el resumen diario del riego efectuado en el día así también como su activación o no, las alertas de notificación vía correo electrónico a ser activadas/desactivadas como: riego (inicio/fin), humedad del suelo (mínimo y máximo), temperatura ambiente (mínimo y máximo), máximo de luz solar y detección de lluvia, la utilización de la cámara web (activo/inactivo) como captura de imagen o de video en tiempo real, además de los parámetros propios para su uso como: tamaño de imagen de captura, tamaño de video y fotogramas por segundos, la activación del uso del correo electrónico para las alertas con sus parámetros necesarios del protocolo smtp como: servidor, usuario, puerto, seguridad y clave, por último si se activa o no el dispositivo así como la configuración cargada se deja desactiva o activa, teniendo en cuenta que solo una configuración puede estar activa.

especies_riegos: para almacenar los riegos realizados durante el día de acuerdo a la especie determinada en la configuración activa con los siguientes datos: fecha/hora del riego (inicio, actualización y finalización), los indicadores de inicio y fin de los datos obtenidos de los sensores como el porcentaje humedad del suelo, los grados celsius de temperatura ambiente y el porcentaje de luz solar, detección de lluvia, la duración en minutos del riego y el volumen de agua suministrado.

historicos_motivos: para almacenar los motivos de eventos que se registrarán en la entidad históricos, los cuales se componen de código con descripción predefinida por el autor.

historicos: para almacenar los eventos registrados (logs) por el dispositivo en cuanto a la: fecha hora, tipo de evento, detalle adicional del evento y si la misma fue notificado, los históricos son disparados en forma automática según el evento ocurrido en el dispositivo.

tarifas_agua: para almacenar la tarifa del costo del agua en el tiempo mediante: la fecha de inicio y fin con su respectiva relación guaraníes a razón de metros cúbicos (Gs/m³), la misma es utilizada al momento de visualizar los reportes generados por la interfaz del usuario para cuantificar la inversión en guaraníes en el consumo del agua.

usuarios: para almacenar los usuarios para el dispositivo mediante: nombres, apellidos, usuario, clave con cifrado MD5, correo electrónico, determinar si el usuario tiene un perfil de administrador o de invitado y si el mismo se encuentra activo, en esta parte todos los usuarios activos recibirán los correos electrónicos de notificación de alertas según activación y ocurrencia como también del resumen diario de riego realizado en el día según el horario fijado.

Una vez terminado el diseño del diagrama de entidad relación se optó por SQLite como sistema de gestión de bases de datos relacional compatible con ACID, de acuerdo a la teoría investigada por el autor, SQLite implementa un manejador de base de datos SQL embebido que facilita su utilización porque no inicia un servicio en segundo plano sistema operativo del dispositivo, independiente de la aplicación, pudiéndose enlazar directamente a código fuente o bien utilizarse en forma de librería ahorrando recursos en el uso de la memoria y del procesador, destacando también su completo soporte para tablas e índices en un único archivo por base de datos, soporte transaccional, escaso tamaño (unas 25 mil líneas de código C) y su completa portabilidad.

La herramienta convenida para trabajar con SQLite fue SQLiteStudio en su distribución de Windows de 32-bits (<https://sqlitestudio.pl/files/sqlitestudio3/complete/win32-inst/InstalSQLite Studio-3.2.1.exe>) por ser una aplicación enteramente gráfica facilitando la curva de aprendizaje de este lenguaje SQL, también se utilizó la versión sqlite3 Command Line Interface

(CLI) instalado en el dispositivo en su sistema operativo Linux OpenWrt para acceder a la base de datos mediante terminal por línea de comandos en forma remota por comando ssh. En el siguiente enlace se presentan íntegramente el script de creación de las tablas de la base de datos denominado sistriego_db

A continuación se muestran por orden de creación: el diagrama entidad relación, la herramienta SLQiteStudio con las tablas creadas, un script de muestra de la tabla especies y los registros cargados en tabla motivos de históricos (historicos_motivos).

Figura 59: Diagrama entidad relación para el dispositivo. Fuente: Autor.

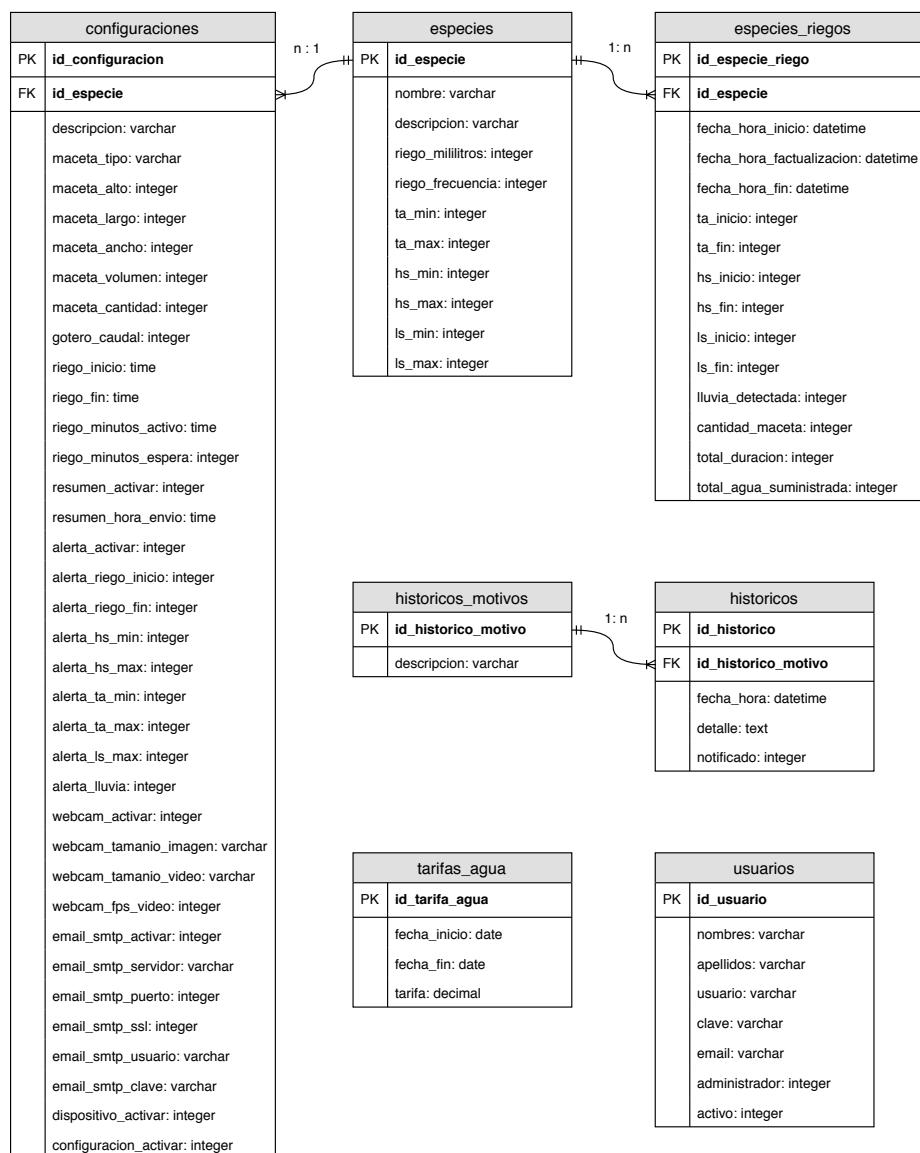


Figura 60: Herramienta administrador de base de datos SQLiteStudio. Fuente: Autor.

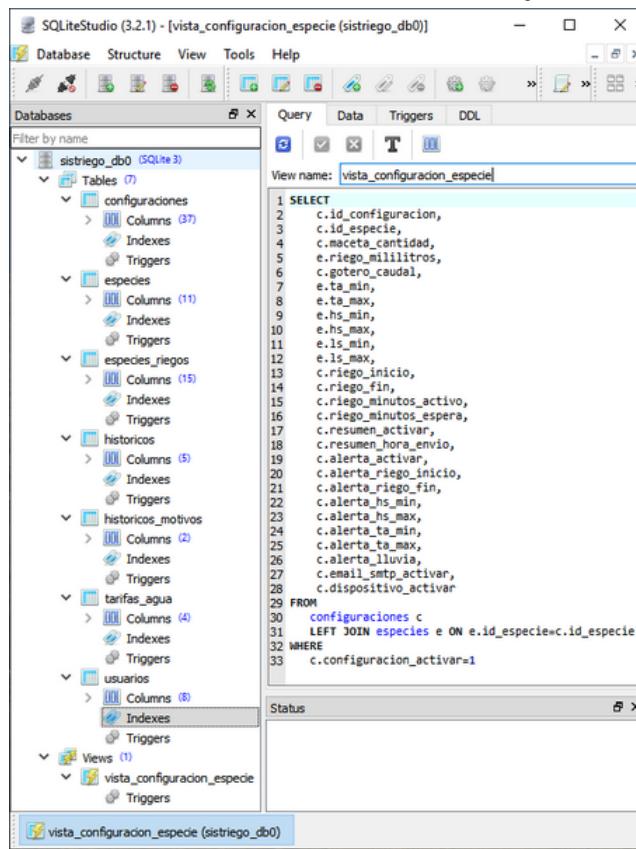


Figura 61: Bloc de notas con Script SQL de muestra de la tabla especies. Fuente: Autor

The screenshot shows a note pad titled 'scriptBaseDatosSistRiego_sistriego_db.txt: Bloc de notas'. The content is a SQL script for creating the 'especies' table. The table has an integer primary key 'id_especie' and various other columns like 'nombre', 'descripcion', and several time-related columns ('ta_min', 'ta_max', 'hs_min', 'hs_max', 'ls_min', 'ls_max'). A unique index 'id_especie_pk' is also created on the 'id_especie' column.

```

-- BASE DE DATOS SQLITE3, CREAR CON EL NOMBRE DE sistriego_db
-- SCRIPT SQL DE TABLAS PARA sistriego_db

-- ****ESPECIES DE CULTIVO
CREATE TABLE especies (
    id_especie      INTEGER CONSTRAINT id_especie_pk PRIMARY KEY ASC ON CONFLICT ROLLBACK AUTOINCREMENT
        UNIQUE
        NOT NULL,
    nombre          VARCHAR NOT NULL,
    descripcion     VARCHAR NOT NULL,
    riego_millilitros INTEGER NOT NULL,
    riego_frecuencia INTEGER NOT NULL,
    ta_min          INTEGER NOT NULL,
    ta_max          INTEGER NOT NULL,
    hs_min          INTEGER NOT NULL,
    hs_max          INTEGER NOT NULL,
    ls_min          INTEGER NOT NULL,
    ls_max          INTEGER NOT NULL
);
CREATE UNIQUE INDEX id_especie_pk ON especies (
    id_especie
);
  
```

Tabla 31: Datos cargados en la tabla historicos_motivos. Fuente: Autor

TABLA: historicos_motivos	
ID	DESCRIPCIÓN
1	DISPOSITIVO INICIADO
2	DISPOSITIVO ACCEDIDO
3	DISPOSITIVO FECHA/HORA ACTUALIZADO
4	DATOS CONFIGURACIONES ABM
5	DATOS ESPECIES ABM
6	DATOS USUARIOS ABM
7	RIEGO ACTIVADO
8	RIEGO FINALIZADO
9	CAPTURA DE IMAGEN REALIZADO
10	VIDEO EN LÍNEA ACTIVADO
11	TEMPERATURA MÁXIMA DETECTADA
12	TEMPERATURA MÍNIMA DETECTADA
13	HUMEDAD SUELO MÁXIMA DETECTADA
14	HUMEDAD SUELO MÍNIMA DETECTADA
15	ILUMINACIÓN MÁXIMA DETECTADA
16	LLUVIA DETECTADA
17	RESUMEN DIARIO
18	BORRADO FILTRADO HISTÓRICOS
19	BORRADO TOTAL HISTÓRICOS
20	DATOS TARIFAS AGUA ABM

Programación del firmware de control automatizado. El firmware de control automatizado fue programado mediante el IDE de desarrollo Arduino en su versión 1.8.10. Por tanto se creó el sketch o boceto con el nombre de “SistRiegoUtic2019.ino” y partiendo del diagrama de flujo del firmware (Ver Figura 48) se fue programando cada rutina del algoritmo como si fuese una receta de cocina, respetando la disposición de pines de la Tabla 29 con una correcta identación del código para facilitar su entendimiento, además se incluyeron las librerías necesarias para ciertos módulos electrónicos, además se crearon funciones específicas para sub-procesos, se agregaron comentarios y por cada cierto tiempo se compilaba para detectar alguna inconsistencia en el programa, una vez que se tenía una parte hecha para cada proceso del dispositivo se compilaba y se subía el código fuente a la placa Arduino Yún para verificar su comportamiento y operación utilizando una herramienta de comunicación serie mediante el puerto USB, provistas por la propia IDE de desarrollo, para visualizar salidas de impresión de control y con ello depurar el código resultante.

Se podrá visualizar en forma integra el sketch creado “SistRiegoUtic2019.ino” en el enlace <https://github.com/geab1976/TesisUtic2019/blob/master/Software/Firmware/SistRiegoUtic2019.ino>. En la siguiente figura se aprecia el IDE de desarrollo Arduino con el sketch del firmware final.

Figura 62: IDE de desarrollo Arduino con el sketch del firmware. Fuente: Autor

The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** SistRiegoUtic2019 Arduino 1.8.10
- Menu Bar:** Archivo, Editar, Programa, Herramientas, Ayuda
- Code Area:**

```

1 /*
2 DOCUMENTO : SistRiegoUtic2019.ino
3 CREADO POR : LIC. GUSTAVO ELOY ALCARAZ BOGADO
4 CREACION : 23/08/2018 04:34:56 PM
5 MODIFICACION : 14/06/2019 20:10:00 PM
6 MODIFICACION : 30/10/2019 22:18:00 PM
7 DESCRIPCION : SISTEMA DE CONTROL AUTOMÁTICO PROGRAMABLE
8 PARA EL RIEGO POR GOTEO DE UN HUERTO URBANO EN
9 MACETAS
10 =====
11 La presente investigación tecnológica se enfocará en crear un
12 dispositivo electrónico que podrá automatizar, configurar y
13 monitorear el proceso de riego por goteo de una huerta urbana
14 en macetas empleando un actuador (electroválvula) controlada
15 por una unidad de control (plataforma de hardware abierto con
16 microcontrolador) mediante sensores de: temperatura ambiental,
17 iluminación, humedad del suelo y lluvia, y por último
18 interactuar en forma inalámbrica (WIFI) mediante una interfaz
19 gráfica multiplataforma (WEB), teniendo como finalidad
20 primordial la racionalización en el uso del suministro de
21 agua según las necesidades de las plantas en el proceso de
22 riego.
23 =====
24 */
25
26 //LIBRERIAS NECESARIAS PARA EL FUNCIONAMIENTO
27 #include <Bridge.h>
28 #include <BridgeServer.h>
29 #include <BridgeClient.h>
30 #include <DHT.h>
31 #include <avr/wdt.h>
32 #include <Process.h>
33 #include <Wire.h>
34 #include <SPI.h>
35 #include "RTClib.h"
36 #include <Watchdog.h>
37 /*
38   ESPECIFICACIONES DE ENTRADAS LÓGICAS Y ANALÓGICAS UTILIZADAS
39 =====
40 SCL -> D7  DS1307  (encontrar en Detall

```
- Bottom Status Bar:** Arduino Yún en COM9

Creación del programa externo para el servicio de telemetría. El programa para el servicio de telemetría, es decir, para dejar de manera accesible los datos obtenidos del ambiente físico por la unidad de control (Arduino Yún) mediante los sensores utilizados a la interfaz gráfica para el usuario, se optó por emplear el lenguaje de programación Python debido a que el mismo se puede instalar en el sistema operativo Linux OpenWrt, además de mencionar en la parte teórica su gran simpleza, cantidad de librerías, robustez y consumo mínimo de recursos haciéndolo indefectiblemente adecuado para el propósito pensado, los ficheros de Python están identificados por la extensión “.py” y su sintaxis emplea solo la identación para agrupar bloques de códigos.

Partiendo del diagrama de flujo del programa externo para servicio de datos de sensores (Ver Figura 49) se observó la necesidad de publicar los valores obtenidos de los sensores para la interfaz gráfica del usuario y así evitar consultar directamente al microcontrolador, por tal motivo opté por acceder a los mismos mediante consultas REST o servicios RESTfull que funcionan como una interfaz entre sistemas que utiliza el protocolo HTTP para obtener datos o generar operaciones sobre esos datos en todos los formatos posibles y devolverlo como notación JSON, de ésta manera se tiene un protocolo cliente/servidor sin estado, utilizando la

operación POST e identificador único de recurso (URI) para cada petición de datos de sensores.

En el firmware creado se empleó una librería puente llamada “brigde” para que puedan comunicarse entre sí el microcontrolador ATmega32u4 y el sistema operativo Linux OpenWrt (microprocesador) del Arduino Yún, con el fin de que a medida que el dispositivo iteraba su ciclo este actualizaba los valores obtenidos de los sensores mediante una estructura clave/valor, dejando disponible y actualizado al microprocesador sin afectar el funcionamiento del microcontrolador por peticiones recurrentes, y en este punto el lenguaje de programación Python también incorpora una librería para recuperar esa estructura de datos y con ello poder actuar de intermediario entre el dispositivo y la interfaz gráfica para el usuario (PHP), evitando la interacción directa entre ellos. En tal sentido para que también el programa actué como consulta REST se utilizó la librería del Python denominado Bottle que es un servidor web minimalista, liviano, rápido, fácil de usar e ideal para crear servicios RESTful (ver Tabla 30).

El programa creado para la funcionalidad se denomina “datos_sensores.py”, el mismo se puede visualizar en el siguiente enlace https://github.com/geab1976/TesisUtic2019/blob/master/Software/ProgramasPy/datos_sensores.py.

Este programa se ejecuta una sola vez en el firmware quedando como servicio en segundo plano en el sistema operativo Linux (ver código fuente de firmware), para programar se utilizó la herramienta Visual Studio Code.

Figura 63: Puente entre Microcontrolador ATmega32u4 y Linux OpenWrt del Arduino Yún.
Fuente: Autor

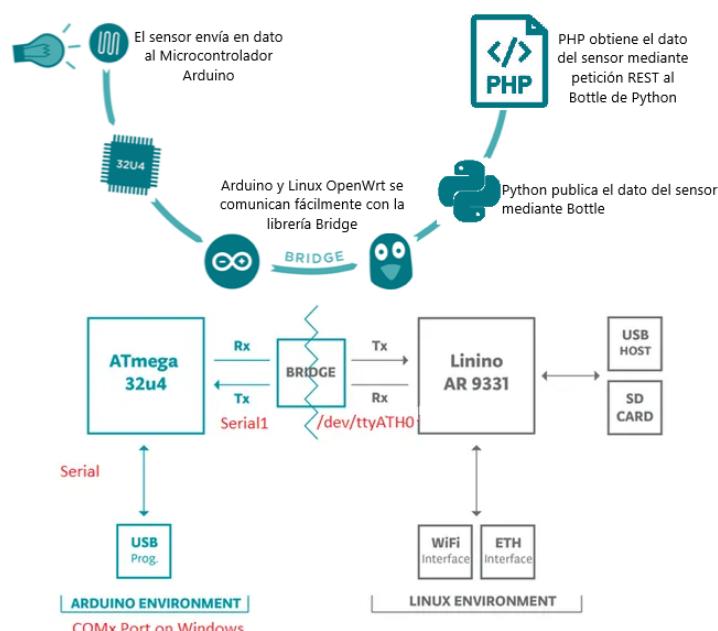


Figura 64: datos_sensores.py (código fuente). Fuente: Autor

The screenshot shows a Visual Studio Code interface with the file 'datos_sensores.py' open. The code is a Python script that imports the 'bridgeclient' module from the 'bridge' package and defines a route handler for '/status'. The handler performs a series of GET requests to a 'bridgeclient' object to retrieve sensor values. The code is color-coded for syntax, and the interface includes a sidebar with icons and a status bar at the bottom.

```

1  #!/usr/bin/python
2  # -*- coding: utf-8 -*-
3  import sys
4  sys.path.insert(0, '/usr/lib/python2.7/bridge')
5  from bridgeclient import BridgeClient as bridgeclient
6
7  sys.path.insert(0, '/usr/lib/python2.7/bottle')
8  from bottle import run, route, get, post, request
9
10 bc = bridgeclient()
11
12 @route('/status')
13 def status():
14     try:
15         fhi    = bc.get("FHIkey")
16         ciclo = bc.get("CICkey")
17         fh    = bc.get("FHkey")
18         hs    = bc.get("HSkey")
19         ha    = bc.get("HAkey")
20         ta    = bc.get("TAkey")
21         luz   = bc.get("LUZkey")
22         lluvia = bc.get("LLUkey")
23         riego = bc.get("ENCkey")
24         ca    = bc.get("CAkey")
25         va    = bc.get("VAkey")
26
27     except:
28         return "Error en la ejecución del script"
29
30     return "OK"
31
32 if __name__ == '__main__':
33     run(host='0.0.0.0', port=8080)

```

Creación del programa externo para el servicio de notificación de alertas. En el programa del firmware creado se observa que se ejecuta un segundo programa externo para servicio de notificaciones de alertas (Ver Figura 48 y código fuente del firmware creado) y analizado el diagrama de flujo correspondiente al mismo (Ver Figura 50), se estipuló que se haría las notificaciones mediante vía correo electrónico, debido a que la placa Arduino Yún tiene la capacidad de conectarse y utilizar internet para actualizar sus paquetes de programas, motivo suficiente para utilizar el protocolo SMTP. Por tanto se realiza una consulta a la base de datos (Ver Tabla 58) creada con respecto a la configuración activa para obtener las alertas activas, los datos sobre los parámetros necesarios para usar el protocolo SMTP, los datos de especies en cuanto a rango mínimos y máximo permitido de los sensores para disparar las alertas y los datos de riegos iniciados o terminados en el día y de acuerdo a las notificaciones realizadas registra cada evento como histórico (logs).

El programa creado para la funcionalidad se denomina “notificaciones_alertas.py”, el mismo se puede visualizar en el siguiente enlace https://github.com/geab1976/TesisUtic2019/blob/master/Software/ProgramasPy/notificaciones_alertas.py.

Este programa se ejecuta una sola vez en el firmware quedando como servicio en segundo plano en el sistema operativo Linux, para programar se utilizó también la herramienta Visual Studio Code.

Figura 65: notificaciones_alertas.py (código fuente). Fuente: Autor

```

notificaciones_alertas.py > ...
1  #!/usr/bin/python
2  # -*- coding: utf-8 -*-
3  import os
4  import sys
5  sys.path.insert(0, '/usr/lib/python2.7/bridge/')
6  from bridgeclient import BridgeClient as bridgeclient
7  import time
8  import datetime
9  import sqlite3
10 from sqlite3 import Error
11 import smtplib
12 from email.mime.multipart import MIMEMultipart
13 from email.mime.text import MIMEText
14 from email.MIMEImage import MIMEImage
15 import mimetypes
16 import httplib
17 import urlparse
18 import urllib
19
20 bc= bridgeclient()
21 fh= "2019-01-01 00:00:00"
22 hs= 0
23 ha= 0
24 ta= 0
25 ls= 0
26 llu= "NO"

```

Python 2.6.0 64-bit Arduino Yún COM22 1

Creación del programa externo de estrategia de control del riego. Analizando el diagrama de flujo del programa externo para la estrategia de control del riego (Ver Figura 51) se necesitó que el firmware envíe como argumentos los datos obtenidos por los sensores al programa directamente y que el mismo se repita a cada ciclo, aproximadamente cada 3 segundos itera la función loop() del firmware creado para ejecutar este programa de estrategia de control del riego, con el fin de evitar consumo innecesario de recursos del hardware al consultar al programa de servicios REST del programa externo datos_sensores.py, ya con los datos recibidos el programa sigue una secuencia de comparaciones para determinar si las condiciones ambientales requiere el riego a partir de los registros de riego en el día y para poder comparar necesita obtener de la base de datos de sistriego_db los datos contenidos en configuraciones, especies y riegos realizados en el día.

La lógica empleada en la estrategia de control se basa en las siguientes comparaciones para Activar o Desactivar: por indicadores ambientales como: los rangos mínimos y máximos permitidos por especie por las condiciones de Temperatura Ambiente, Humedad del Suelo, Luz Solar y Detección de Lluvia, y por riegos realizados en el día como: Configuración y Dispositivo Activos, Horario de Riego Permitido, Existe Registro de Riego, Tiene Fecha/Hora Fin en el Registro de Riego, Indicadores Ambientales Permitidos, Dentro del Tiempo Activo de Riego Permitido, Pasado el Tiempo Espera entre Riego Permitido, Menor al Volumen por Riego Permitido y Menor al Volumen Total por Día Permitido, de acuerdo a los resultados lógicos de inserta, actualiza o cierra el registro de riego en el día, a través de la Tabla 32 se

visualiza las comparaciones para la estrategia de control. En caso de inicio y fin de riego en el día se genera un histórico por cada evento, y a partir de estos registros se verifica si se encuentra activo la alerta por inicio o fin de riego para notificar la alerta correspondiente.

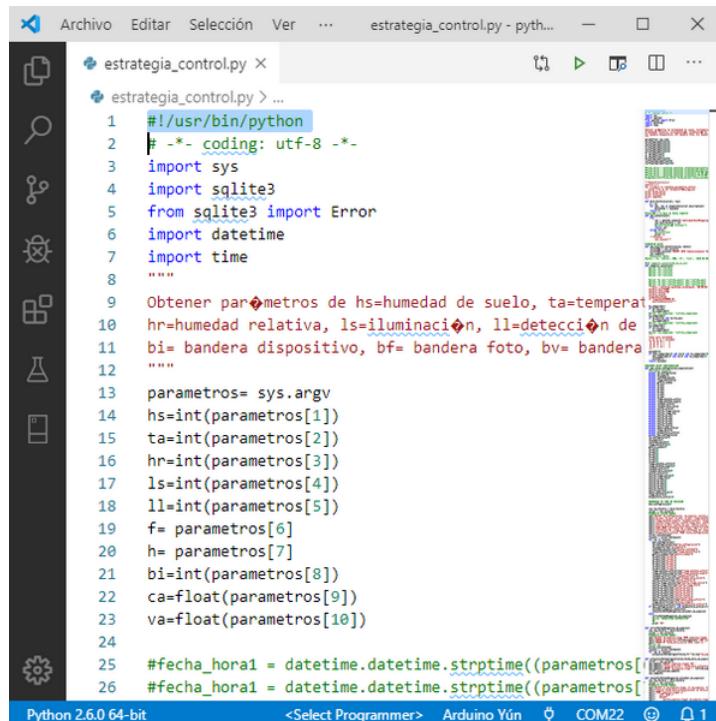
Este programa externo devuelve al firmware cuatro tipo posibles de acciones como se puede apreciar en el diagrama de flujo del firmware correspondiente al sub-proceso de estrategia de control de la Figura 48 .

Tabla 32: Comparativo utilizado por la Estrategia de Control del riego en el día. Fuente: Autor

El programa creado para la funcionalidad se denomina “estrategia_control.py”, el mismo se puede visualizar en el siguiente enlace https://github.com/geab1976/TesisUtic2019/blob/master/Software/ProgramasPy/estrategia_control.py.

Este programa se ejecuta y devuelve la acción por cada iteración del ciclo look() al firmware del dispositivo, para programar se utilizó también la herramienta Visual Studio Code.

Figura 66: estrategia_control.py (código fuente). Fuente: Autor



The screenshot shows a Visual Studio Code interface with the following details:

- Title Bar:** Archivo, Editar, Selección, Ver, ..., estrategia_control.py - pyth...
- Left Sidebar:** Includes icons for file operations (New, Open, Save, Find, Replace, Go To, Settings, Help).
- Central Area:** Code editor containing the Python script 'estrategia_control.py'. The code is as follows:

```

1  #!/usr/bin/python
2  #-*- coding: utf-8 -*-
3  import sys
4  import sqlite3
5  from sqlite3 import Error
6  import datetime
7  import time
8
9  Obtener parámetros de hs=humedad de suelo, ta=temperatura
10 hr=humedad relativa, ls=iluminación, ll=detección de
11 bi=bandera dispositivo, bf=bandera foto, bv=bandera
12 """
13
14 parámetros= sys.argv
15 hs=int(parámetros[1])
16 ta=int(parámetros[2])
17 hr=int(parámetros[3])
18 ls=int(parámetros[4])
19 ll=int(parámetros[5])
20 f= parámetros[6]
21 h= parámetros[7]
22 bi=int(parámetros[8])
23 ca=float(parámetros[9])
24 va=float(parámetros[10])
25 #fecha_hora1 = datetime.datetime.strptime((parámetros[11]), "%Y-%m-%d %H:%M")
26 #fecha_hora1 = datetime.datetime.strptime((parámetros[12]), "%Y-%m-%d %H:%M")

```

- Bottom Status Bar:** Python 2.6.0 64-bit, <Select Programmer>, Arduino Yún, COM22, 1

Creación del programa externo de activación de la cámara web. Analizando el diagrama de flujo del programa externo para utilizar la cámara web (Ver Figura 52) se necesitó que el firmware envíe como argumento el tipo de activación en este caso 1 para captura de imagen o 2 para activación de video en tiempo real (Ver Figura 48 1: captura de imagen, 2: video en tiempo real).

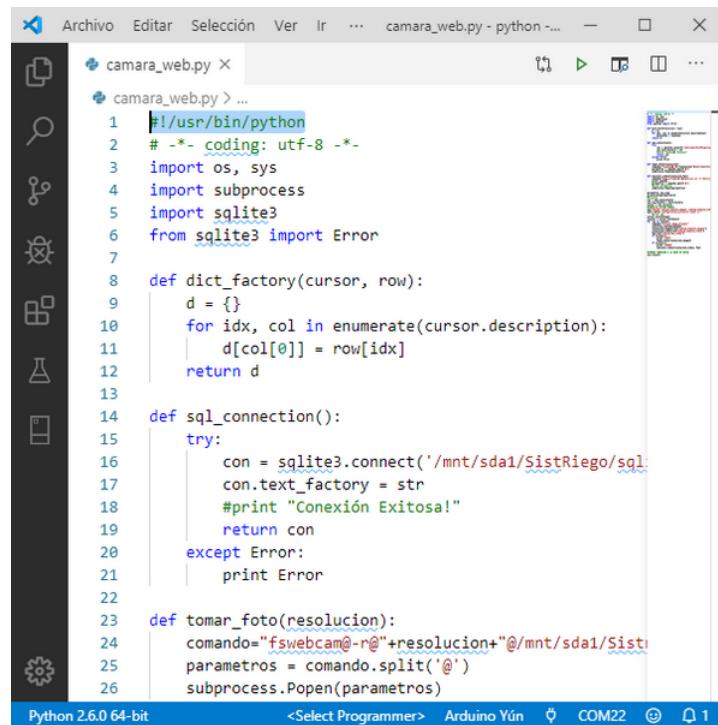
En el caso 1: se utilizó el aplicativo de imagen denominado “fswcam” para capturar imagen que permite configurar opciones generales, de captura y de salida de imagen.

En el caso 2: se utilizó el aplicativo de video denominado “mspj-streamer” para capturar imagen que permite configurar opciones generales, de captura y de salida de imagen.

Tanto el aplicativo “fswcam” y “mspj-streamer” pertenecen a los paquetes opcionales del S.O. Linux OpenWrt ver Tabla 30 (Fila: VIDEO) y Anexo D para mayor detalle.

El programa creado para la funcionalidad se denomina “camara_web.py”, el mismo se puede visualizar en el siguiente enlace https://github.com/geab1976/TesisUtic2019/blob/master/Software/ProgramasPy/camara_web.py. Este programa se ejecuta en forma asíncrona según petición externa al firmware, para programar se utilizó también la herramienta Visual Studio Code.

Figura 67: camara_web.py (código fuente). Fuente: Autor



The screenshot shows a Visual Studio Code interface with the file 'camara_web.py' open. The code is a Python script with the following content:

```
1  #!/usr/bin/python
2  # -*- coding: utf-8 -*-
3  import os, sys
4  import subprocess
5  import sqlite3
6  from sqlite3 import Error
7
8  def dict_factory(cursor, row):
9      d = {}
10     for idx, col in enumerate(cursor.description):
11         d[col[0]] = row[idx]
12     return d
13
14 def sql_connection():
15     try:
16         con = sqlite3.connect('/mnt/sda1/SistRiego/sql')
17         con.text_factory = str
18         #print "Conexión Exitosa!"
19         return con
20     except Error:
21         print Error
22
23 def tomar_foto(resolucion):
24     comando="fswebcam@-r@"+resolucion+"@/mnt/sda1/Sist"
25     parametros = comando.split('@')
26     subprocess.Popen(parametros)
```

The status bar at the bottom indicates "Python 2.6.0 64-bit" and "Select Programmer".

Creación del software de la interfaz gráfica del usuario. Se creó una aplicación multiplataforma web con arquitectura cliente/servidor para el software de la interfaz gráfica del usuario.

Se utilizó una estructura de tres capas:

- Capa del navegador: que implementa las tecnologías HTML, CSS y JavaScript, además de utilizar una biblioteca del JavaScript denominado JQuery Mobile el cual es un framework optimizado para dispositivos táctiles.
- Capa del servidor: que utiliza el lenguaje de programación de código abierto PHP mediante el servidor de aplicaciones uHTTPD que es un servidor web escrito desde cero por los desarrolladores de OpenWrt / LuCI, adecuado para tareas livianas comúnmente utilizadas con dispositivos integrados y una integración adecuada con el marco de configuración (UCI) de OpenWrt, en particular proporciona toda la funcionalidad esperada de los servidores web actuales, en esta parte optó por un patrón de diseño MVC (Modelo Vista Controlador), cuenta también con una función de sesión para mantener identificado al usuario y controlar el tiempo de sesión permitido sin actividad.

- Capa de persistencia: que emplea la base de datos SQLite utilizada por los programas externos creados.

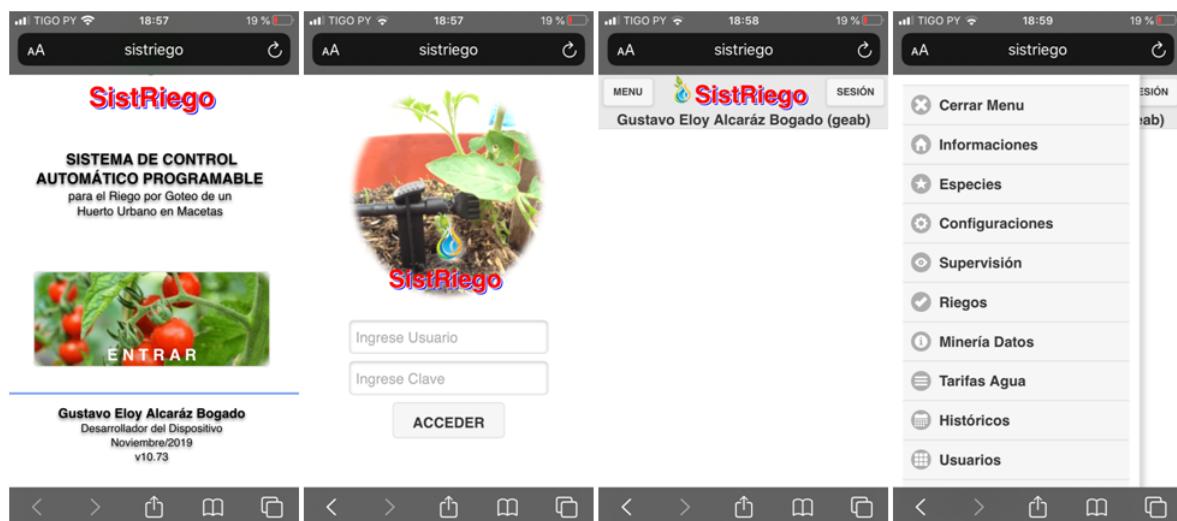
La estructura de archivos es la siguiente: la aplicación web se aloja en una sub-carpeta de nombre “web” dentro de la carpeta de nombre “SistRiego” la cual se encuentra almacenado en la raíz de la tarjeta microSD de 16 GB, dicha sub-carpeta se encuentra organizada de la siguiente manera: una carpeta “css” (hojas de estilos), una carpeta “js” (librerías JavaScript), carpeta “images” (imágenes utilizadas) y en la raíz de la sub-carpeta los archivos de programas hechos en el lenguaje PHP con la siguiente nomenclatura para distinguir: la vista (nombreFuncionalidadHome.php), el modelo (nombreFuncionalidadDatos) y el controlador (nombreFuncionalidadControl).

A continuación se muestran las partes del software de interfaz gráfica del usuario con breves explicaciones e imágenes captadas desde un dispositivo móvil (Iphone SE):

- Presentación (index.php): es la página web de bienvenida de la interfaz gráfica cuya dirección de acceso web es <http://sistriego/web> o la dirección ip fijada al configurar, en mi caso <http://192.168.1.224/web>.

Figura 68: Captura pantalla: presentación, control de acceso, página principal y menú activado.

Fuente: Autor



- Control de Acceso: una vez pulsado en botón de “ENTRAR” en la página de presentación la página web de control de acceso se despliega para que ingresemos el usuario “geab” y la contraseña “321” registrada en la tabla usuarios como muestra de usuario administrador.

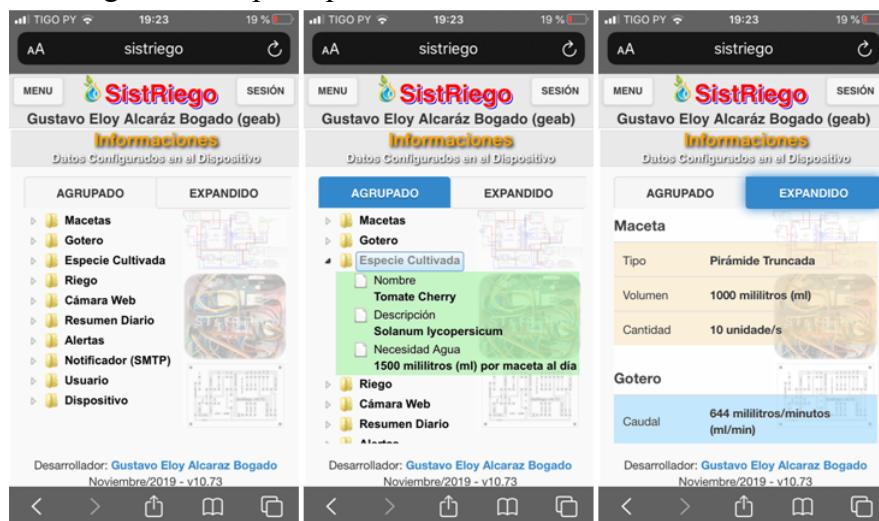
- Principal (paginaPrincipal.php): es la página web con diseño “one page” (una sola página) en donde a medida que una funcionalidad es accionada en el menú ésta es desplegada en la misma página principal.

- Menú de Navegación: el sistema cuenta con dos tipos de menú de navegación para las funcionalidades, el primer menú se ubica al lado superior izquierdo de la página

web, contiene las funcionalidades generales disponibles según el tipo de usuario: el usuario administrador puede visualizar las siguientes funciones: Informaciones, Especies, Configuraciones, Supervisión, Riegos, Minería de Datos, Tarifas Agua, Históricos, Usuarios y Fecha/Hora, mientras que el usuario invitado solo podrá visualizar: Informaciones, Supervisión, Riegos, Minería de Datos y Tarifas Agua. El segundo menú se ubica al lado superior derecho de la página web, conteniendo las funciones de sesión del usuario: Perfil y Cerrar Sesión.

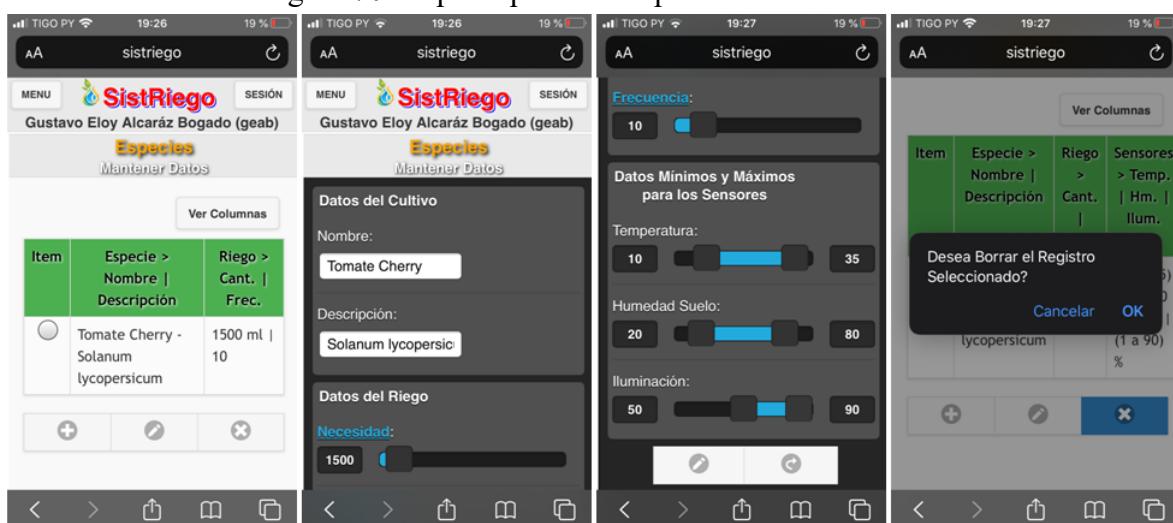
- Informaciones (informacionesHome.php): es la página web que visualiza las configuraciones generales actuales del dispositivo.

Figura 69: Captura pantalla: informaciones. Fuente: Autor



- Especies (especiesHome.php): es la página web en que se mantiene (alta / baja / modificación) los registros con los datos de las especies de cultivos.

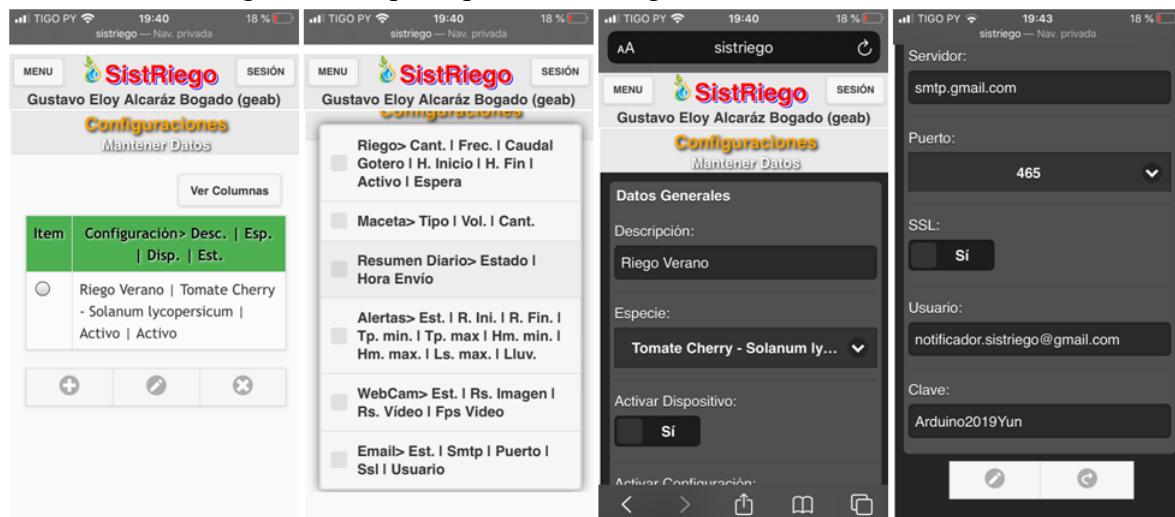
Figura 70: Captura pantalla: especies. Fuente: Autor



- Configuraciones (configuracionesHome.php): es la página web en que se mantiene (alta / baja / modificación) los registros con los datos de las configuraciones personalizada a partir

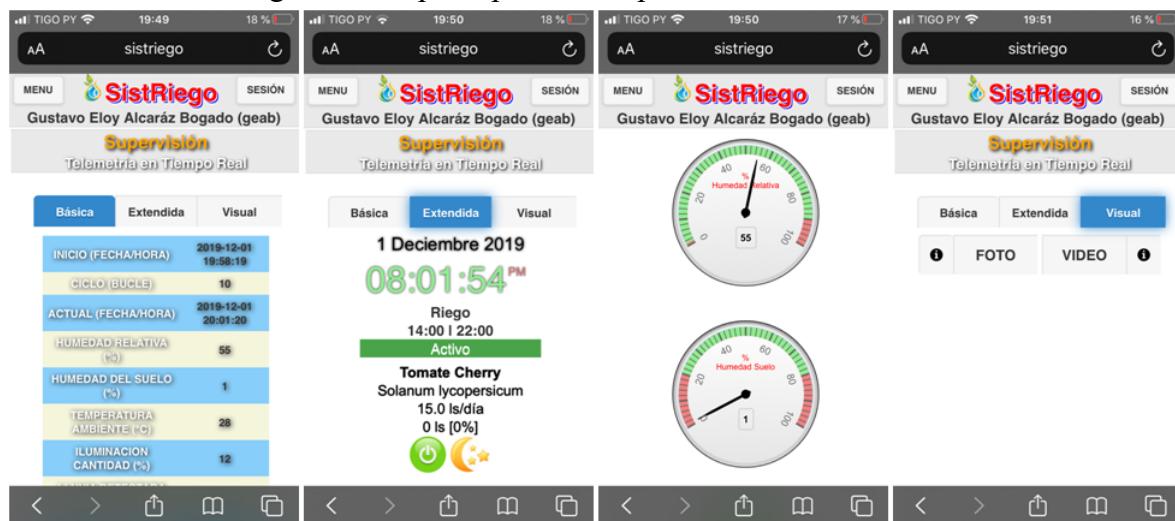
de una especie de cultivo seleccionado, en ella el dispositivo obtiene toda la información de referencia para su funcionamiento de control automatizado como: estrategia de control de riego, notificación de alertas y parámetros de activación de la cámara web.

Figura 71: Captura pantalla: configuraciones. Fuente: Autor



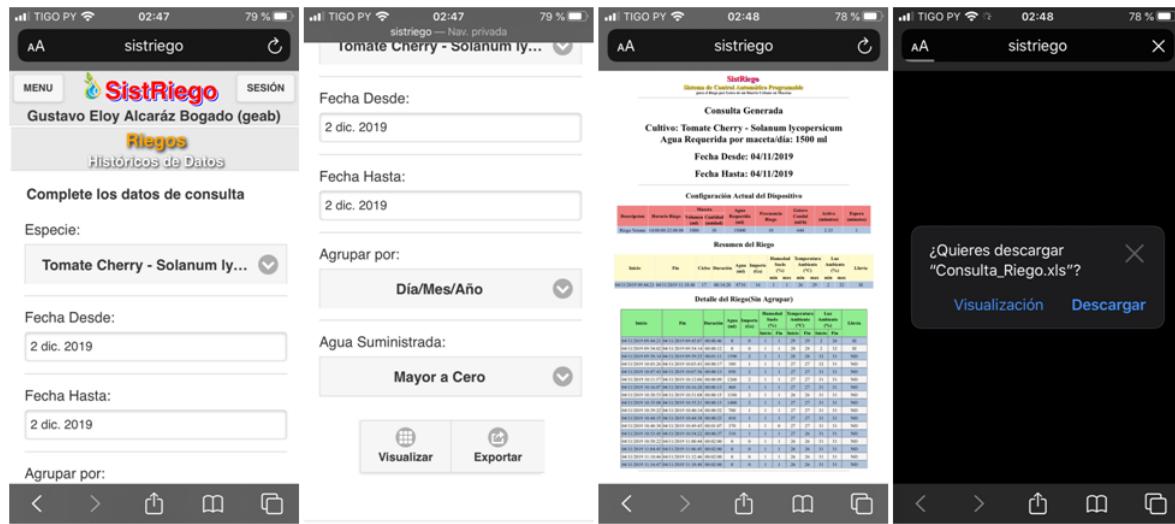
- Supervisión (supervisiónHome.php): es la página web en que se tiene la supervisión del dispositivo en tiempo real tanto de los valores de sensores (telemetría), estado del dispositivo, riegos realizados en el día (volumen y frecuencia), con la opción de visualizar con la cámara web en tiempo real el ambiente físico, ya sea con una captura de imagen o visualización en video.

Figura 72: Captura pantalla: supervisión. Fuente: Autor



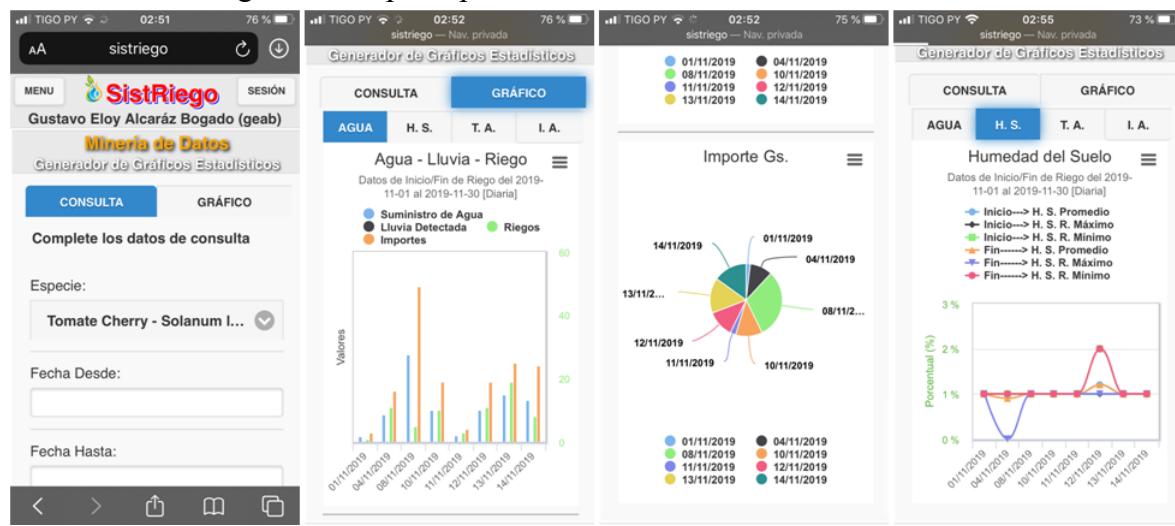
- Riegos (riegosHome.php): es la página web en que se puede obtener el reporte de riegos realizados por: especie, rango de fecha, agrupación de fechas y si fueron activas con/sin agua.

Figura 73: Captura pantalla: riegos. Fuente: Autor



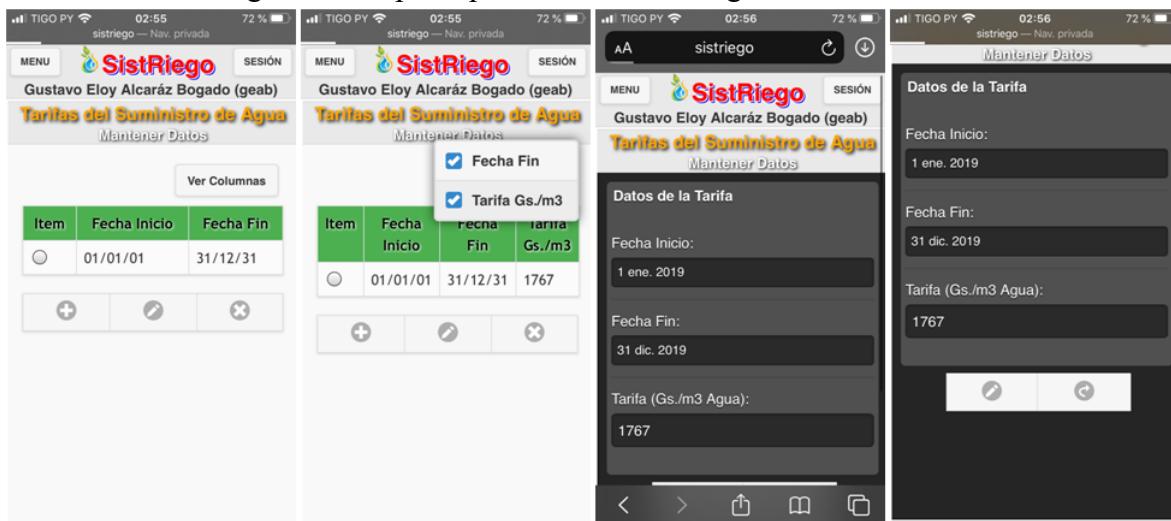
- Minería de Datos (datamineHome.php): es la página web que facilita al usuario a contemplar los datos obtenidos de los riegos e indicadores ambientales en forma de gráficos comparativos, enriqueciendo la obtención de datos para su análisis, es un complemento para el reporte de riegos.

Figura 74: Captura pantalla: minería de datos. Fuente: Autor



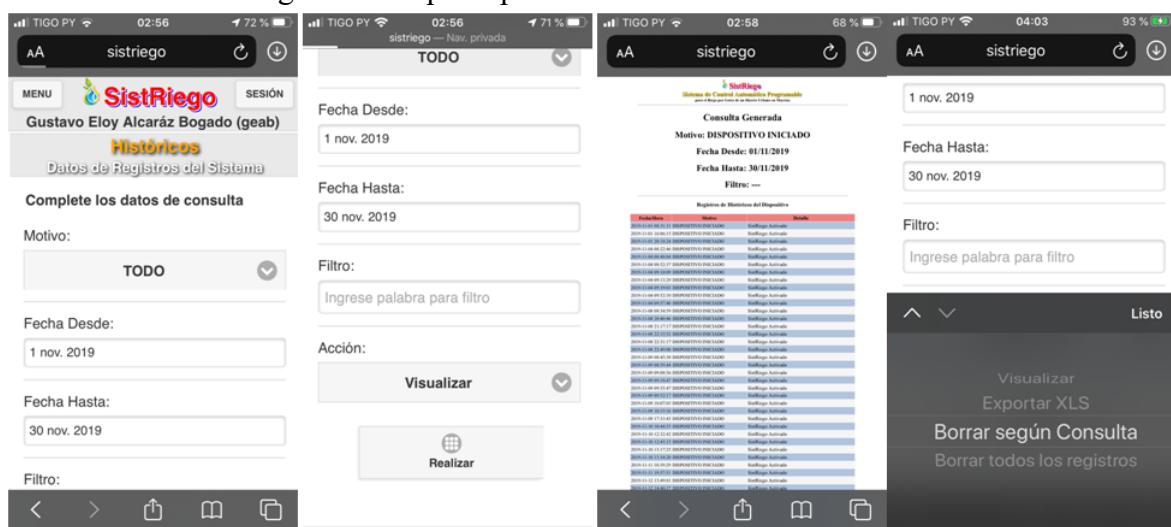
- Tarifas Agua (tarifasHome.php): es la página web en que se mantiene (alta / baja / modificación) los registros con los datos de las tarifas del agua según el periodo existente en relación costo guaraníes sobre metro cúbico (Gs/m3).

Figura 75: Captura pantalla: tarifas de agua. Fuente: Autor



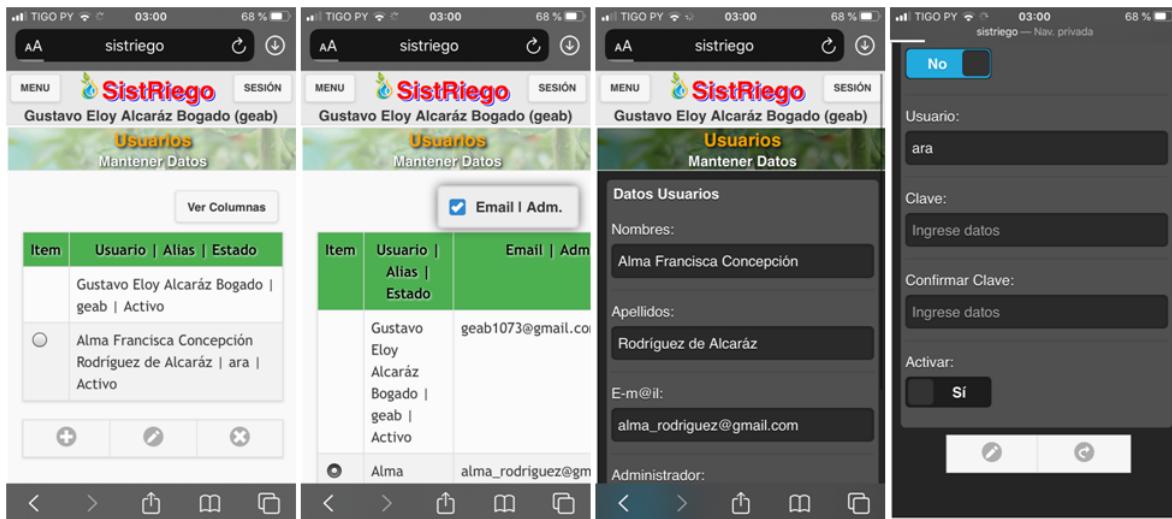
- Históricos (historicosHome.php): es la página web en que se puede obtener el reporte de históricos de eventos ocurridos en el dispositivo (log).

Figura 76: Captura pantalla: históricos. Fuente: Autor



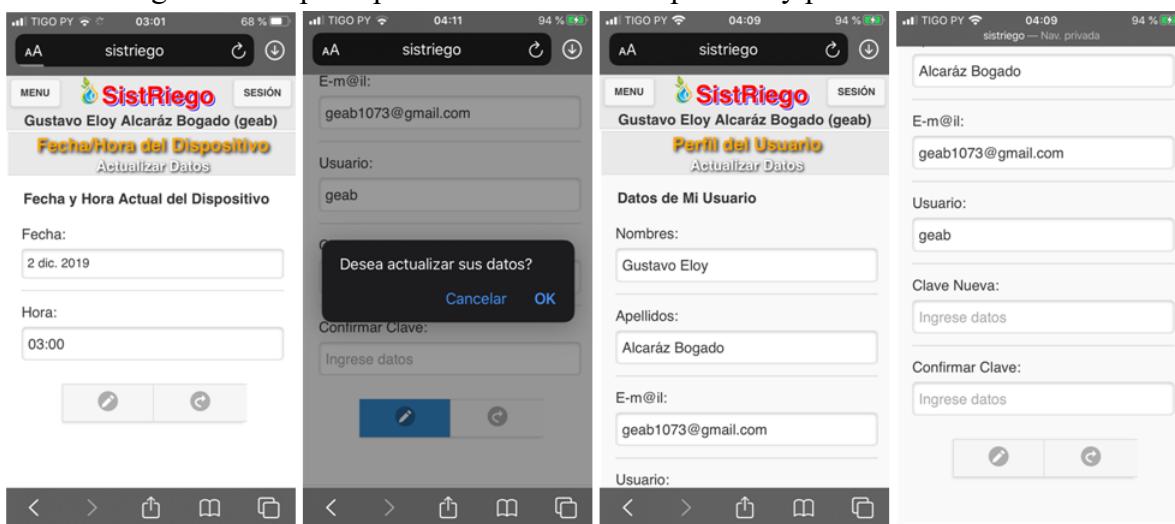
- Usuarios (usuariosHome.php): es la página web en que se mantiene (alta / baja / modificación) los registros con los datos de usuarios del dispositivo, solo el usuario administrador tiene acceso para crear nuevos usuarios y/o desactivarlos.

Figura 77: Captura pantalla: usuarios. Fuente: Autor



- Fecha/Hora (fechaHoraHome.php): es la página web en que se puede actualizar la fecha y la hora del dispositivo.

Figura 78: Captura pantalla: fecha/hora dispositivo y perfil. Fuente: Autor



- Perfil (perfilHome.php): es la página web en que usuario puede cambiar los datos de su perfil de usuario, así como cambiar su clave de acceso.

Todos los artefactos para el software de la interfaz gráfica del usuario fueron programados mediante la herramienta Visual Studio Code, éstos programas se pueden visualizar en el siguiente enlace <https://github.com/geab1976/TesisUtic2019/blob/master/Software/Interfaz/web>.

En la Figura 80 se puede apreciar el esquema de funcionamiento del dispositivo teniendo en cuenta todo lo desarrollado tanto para el hardware y el software del dispositivo.

Figura 79: Código fuentes de la vista, el modelo y el control para usuarios. Fuente: Autor

The figure consists of three vertically stacked screenshots of the Visual Studio Code editor. Each screenshot shows a different file:

- Top Screenshot:** Shows the code for `usuariosHome.php`. The code includes PHP session handling, HTML structure with a table and center, and a script block containing a variable `id_acceso`.
- Middle Screenshot:** Shows the code for `usuariosDatos.php`. This is a PHP script that processes POST data to update a SQLite database. It defines variables for each form field and constructs a SQL query based on the selected option.
- Bottom Screenshot:** Shows the code for `usuariosControl.js`. This is a JavaScript file using jQuery's AJAX functionality to call `usuariosDatos.php` and handle the returned JSON data to update a table.

```

1 <?php
2 session_start();
3 if (!empty($_SESSION['usuario'])) {
4     $lifetime = 1800;
5     setcookie(session_name(), session_id(), time() + $lifetime, "/");
6 }
7 <!DOCTYPE html>
8 <html class="ui-mobile">
9
10 <head>
11     <?php
12         include("head_html.php");
13     ?>
14     <link href="css/estilo_tabla.css" rel="stylesheet" />
15     <script type="text/javascript">
16         var id_acceso = "<?php echo $_SESSION['id']?>";
17     </script>
18 </head>
19
20 <body>
21     <div data-role="page" data-theme="a" id="divGrilla">
22         <div data-role="header" class="sr-usuarios">
23             <center>
24                 <span>

```

```

1 <?php
2 session_start();
3 $lifetime = 1800;
4 setcookie(session_name(), session_id(), time() + $lifetime, "/");
5 //Parámetros enviados
6 $opcion = $_POST["opcion"];
7 $id = $_POST["id"];
8 $nombres = $_POST["nombres"];
9 $apellidos = $_POST["apellidos"];
10 $email = $_POST["email"];
11 $administrador = $_POST["administrador"];
12 $activo = $_POST["activo"];
13 $usuario = $_POST["usuario"];
14 $clave = $_POST["clave"];
15 //Conexión a la base de datos SQLite
16 $db = new SQLite3('/mnt/sda1/SistRiego/sqlite/sistriego_db.db');
17 assert($db);
18 $sql = "";
19 $datos = array();
20 //Selector de casos
21 switch ($opcion) {
22     case "1": //Consultar todos los Datos
23     case "2": //Obtener Datos de un registro
24         $sql = "SELECT id_usuario, usuario, nombres, apellidos, email, adminis

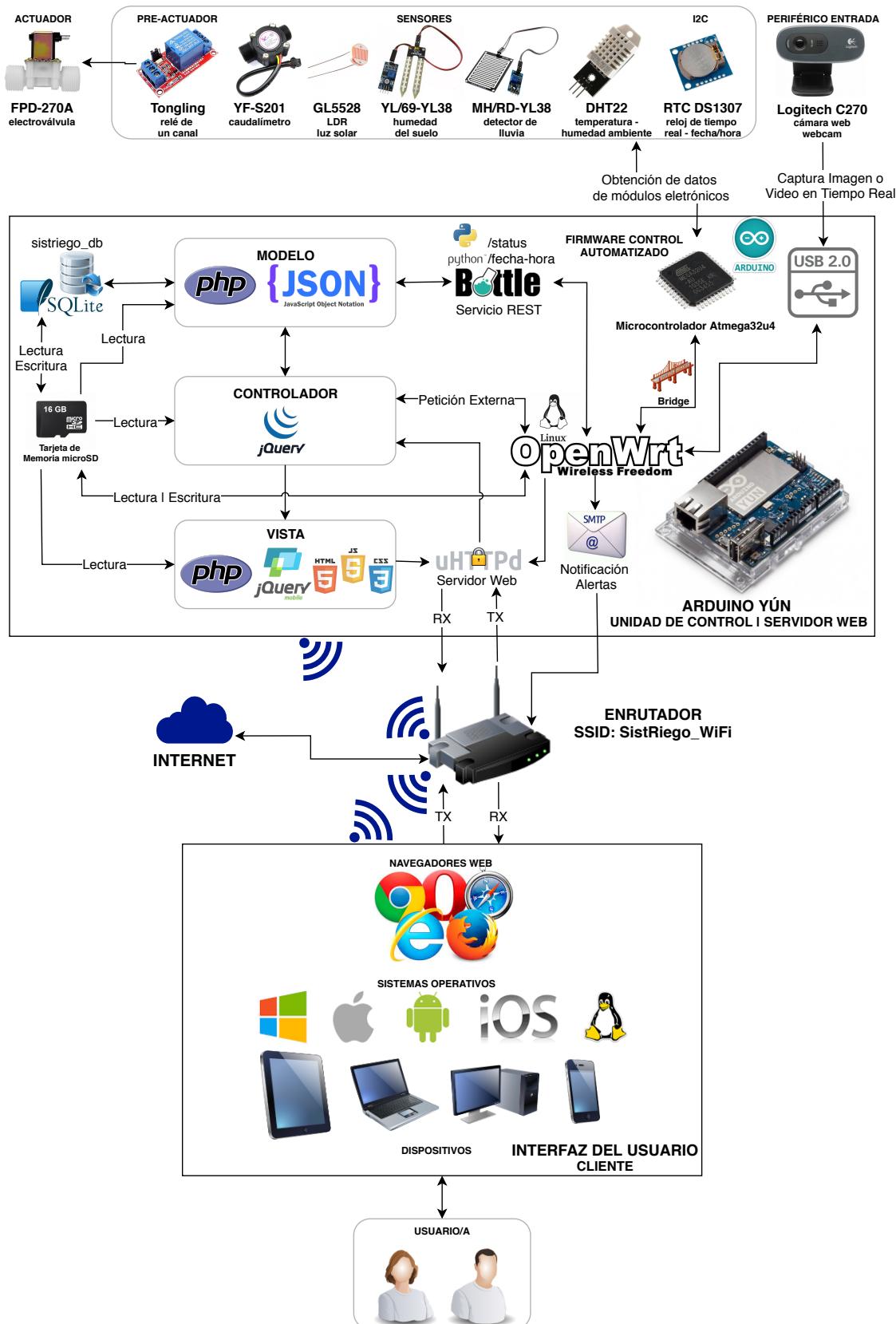
```

```

1 function cargarDatos(opcion) {
2     $.ajax({
3         url: "usuariosDatos.php",
4         data: {
5             opcion: opcion,
6             id: opcion === 1 ? "0" : $("[name=optData]:checked").val(),
7             nombres: opcion === 3 || opcion === 4 ? $("#nombres").val() : "",
8             apellidos: opcion === 3 || opcion === 4 ? $("#apellidos").val() : "",
9             email: opcion === 3 || opcion === 4 ? $("#email").val() : "",
10            administrador: opcion === 3 || opcion === 4 ? $("#administrador").v
11            activo: opcion === 3 || opcion === 4 ? $("#activo").val() : "",
12            usuario: opcion === 3 || opcion === 4 ? $("#usuario").val() : "",
13            clave: opcion === 3 || opcion === 4 ? $("#clave").val() : ""
14        },
15        dataType: "json",
16        method: "POST",
17        success: function(data) {
18            //alert("1");
19            if (opcion === 1) {
20                var tbl = $("#tblDatos tbody").html("");
21                $.each(data, function(id, dato) {
22                    tbl.append("<tr><td>" + (id_acceso != dato.id_usuario ?
23                        "<input type='radio' value='" + dato.id_usuario +
24                        "' name='optData'" : "") + "</td><td>" +

```

Figura 80: Esquema de funcionamiento del dispositivo. Fuente: Autor



Etapa 4: Final. En ésta etapa de se procede a colocar en cajas plásticas externas al hardware del dispositivo, a modo de resguardo y de preparación para las conexiones entre las mismas mediante cables de par trenzado, luego de proteger el hardware se procedió realizar las pruebas y ajustes finales en condiciones reales del ambiente, una vez verificado y convalidado las mismas se procedió a documentar todo lo realizado a través de la redacción del informe final.

Adecuación del hardware del dispositivo en cajas plásticas externas. El hardware del dispositivo final fueron separados en cuatro cajas plásticas de conexión externa utilizadas en las instalaciones eléctricas de marca la Elektron, a continuación se muestran la disposición final en la Tabla 33 y se visualiza las mismas en la Figura 69.

Tabla 33: Disposición del hardware en cajas plásticas externas. Fuente: Autor

Item	Caja	Hardware en su interior	Conexiones	Conecta
A	largo: 15 cm ancho: 15 cm alto: 7 cm forma: rectangular	unidad de Control (Arduino Yún) módulo reloj en tiempo real (RTC DS1307) placa de circuito impreso de conexiones módulo sensor detector de lluvia (YL-38) sensor humedad del suelo (YL-38) indicador luminoso (Led RGB)	Jack RJ45 hembra interno: sensor temperatura y humedad ambiente (DHT22) electrodo sensor humedad de suelo (YL-69) sensor de luminosidad (LDR GL5528) Jack RJ11 hembra interno: electrodo sensor detector de lluvia (MH-RD) Ficha Dupont (hembra 3 pines, macho 1 pin): pre-actuador (relé tongling) tierra común (GND) Conector externo USB hembra: cámara web (Logitech C270) Conector externo USB macho: fuente de 5V 2A	B y C
B	diámetro: 8 cm alto: 5 cm forma: circular	sensor temperatura y humedad ambiente (DHT22) electrodo sensor humedad de suelo (YL-69) sensor de luminosidad (LDR GL5528) electrodo sensor detector de lluvia (MH-RD)	Jack RJ45 hembra interno: sensor temperatura y humedad ambiente (DHT22) electrodo sensor humedad de suelo (YL-69) sensor de luminosidad (LDR GL5528) Jack RJ11 hembra interno: electrodo sensor detector de lluvia (MH-RD)	A
C	largo: 10 cm ancho: 10 cm alto: 7 cm forma: rectangular	pre-actuador (relé tongling) sensor de caudal de agua (YF-S201) tierra común (GND) indicador luminoso (Led Rojo)	Ficha Dupont (hembra 3 pines, macho 1 pin): pre-actuador (relé tongling) sensor de caudal de agua (YF-S201) tierra común (GND) Ficha Dupont (hembra 5 pines, macho 1 pin): sensor de caudal de agua (YF-S201) electroválvula (FPD-270A) Jack hembra fuente DC: fuente de 9V 1A	A y D
D	diámetro: 8 cm alto: 5 cm forma: circular	sensor de caudal de agua (YF-S201) electroválvula (FPD-270A) indicador luminoso (Led Verde)	Jack RJ45 hembra externo: sensor de caudal de agua (YF-S201) electroválvula (FPD-270A)	C

Figura 81: Cajas plásticas externas terminadas para el dispositivo. Fuente: Autor



A



B



C



D



Pruebas y ajustes finales. Primeramente se calibraron los goteros del sistema de riego, el gotero utilizado tiene la facilidad de ajustar el pico inyector mediante giros a la derecha (más flujo) o a la izquierda (menos flujo), además de realizar un sonido del tipo clic en cierto momento del giro, los cinco goteros fueron puestos en el punto de cierre, luego se conectó el gotero a un micro tubo de 1 metro de longitud, este al reductor de presión de agua de 1 bar, la cual se encuentra por el pico de la canilla de suministro de agua, luego se procedió a abrir la llave de agua y se fue ajustando el gotero hasta obtener un flujo de goteo continuo contabilizando los giros necesarios a la derecha con los sonidos de clic, obteniéndose 12 clic, luego fueron replicados los demás goteros con los 12 clic audibles, una vez determinado la intensidad del goteo, se procedió a cargar un vaso de medida de 500 ml con uno de los goteros previamente ajustados, en este paso se cronometró el tiempo de llenado del vaso, para determinar el caudal del gotero, en mi caso luego de realizar 5 pruebas se llegó a que el gotero tenía un caudal aproximado de 500 mililitros a 0.776 minutos (46,78 segundos), y por regla de tres simple se tuvo que 644 mililitros de agua corresponden a 1 minuto. Este procedimiento se hizo con los demás goteros para dejarlo listo para las pruebas, con ésto se obtuvo por teoría la capacidad de suministro de agua de los goteros en 1 minuto, de acuerdo a la presión del agua en ese momento.

Posteriormente el siguiente elemento a ser calibrado fue el sensor de caudal de agua del dispositivo, utilizando las referencias técnicas del fabricante, consistente en la fórmula de caudal del modelo YF-S201, donde estipula que el flujo del agua en litros por minuto es igual a las pulsaciones del sensor (Hz) sobre el valor 7.5, donde 7.5 es la constante de fábrica, para ajustar esta constante, se procedió a configurar el dispositivo para que entregue en intervalos de riego el volumen de 500 mililitros, entonces el dispositivo accionará la electroválvula técnicamente hasta completar 500 mililitros, para ello se registró el valor del volumen de agua por día requerido para el cultivo a 2,5 litros (2.500 mililitros), con una frecuencia de 5, es decir, que por cada riego el cultivo va a recibir 500 mililitros, con este nuevo dato, se procedió a ajustar la configuración del riego poniendo en tiempo de espera entre riego de 5 minutos, para tener tiempo de vaciar el vaso y volver a observar la experiencia, además se agregó el valor del caudal del gotero obtenido, que fue de 644 mililitros por minuto, con este dato el tiempo de encendido de la electroválvula va hasta 7,7 minutos. Una vez fijado todos los parámetros se borraron de la base de datos los datos del riego realizado en el día para que el dispositivo vuelva a reiniciarse el riego, con esto el dispositivo se accionó automáticamente por ser el primer intervalo de riego en el día, previamente se conectó a la electroválvula la manguera de suministro de agua, el reductor de presión de agua de 1 bar por la salida de 1/2" del caudalímetro, del reductor de presión al micro tubo, del micro tubo al gotero calibrado, y éste al vaso de medida de 500 mililitros, con ésto se cronometró cuantos minutos necesitaba el dispositivo para completar 500 mililitros, arrojando los siguientes resultados, el volumen registrado por el dispositivo era superior con respecto al verificado teniendo entre cada intervalo una cantidad aproximada de 315 mililitros de más, es decir, que para 500 mililitros reales verificados el dispositivo cuantificaba 815 mililitros, entonces, se ajustó el factor 7,5 a 4,6 teniendo en cuenta la regla de tres simple, si 815 ml es a

un factor de 7,5 que factor sería a 500 ml ($500 \times 7.5 / 815 = 4.601$), se corrigió el factor del caudal en el firmware con 4,6 y se sube el sketch compilado al dispositivo, luego se volvió a repetir la experiencia arrojando una exactitud de casi +/- 5%.

Paso se seguido para las pruebas y ajustes iniciales se utilizó un huerto urbano compuesto por 5 macetas de 14 litros con cultivos de tomates con dos meses y medio de haber germinado la semilla, ya en etapa de floración, los mismos se encuentran alineados en una sola fila, cada uno con un gotero calibrado, luego el dispositivo fue puesto sobre una mesita enfrente a las macetas, al igual que otra mesita para posar la computadora portátil y verificar los datos arrojados en la experiencia del funcionamiento, por último la electroválvula fue conectada a la fuente de suministro de agua con la llave abierta y la caja de sensores montada sobre el borde de la maceta central como punto de referencia, además el electrodo del sensor de humedad del suelo fue posicionado a 10 cm del tallo del cultivo.

Una vez puesta en funcionamiento el dispositivo, se fueron comprobaron las funcionalidades creadas mediante la interfaz gráfica del usuario, se verificaron que los sensores de; temperatura y humedad ambiente, humedad del suelo, luz solar y el detector de lluvia operaron dentro del margen de lo esperado, además se comprobó que desconectando el dispositivo y volviendo a encender no perdía la fecha y hora registrado de el primer día de configuración gracias a su módulo independiente de reloj en tiempo real RTC DS1307, el cual posee una pila plana de litio para mantener la configuración.

En cuanto al riego se certificó los casos enunciados en la Tabla 32 y el correcto registro en la base de datos del SQLite.

En cuanto a las funcionalidades de la cámara web se comprobó que a una configuración intermedia de resolución, su desempeño era óptimo, teniendo en cuenta la cantidad de procesos activos en el dispositivo, además de comprobar su correcto funcionamiento en diferentes equipos clientes.

En relación a las alertas del dispositivo vía correo electrónico se comprobaron su funcionamiento de acuerdo a lo programado, sólo se hicieron ajustes en los detalles visuales de las notificaciones enviadas por cada caso.

En cuanto a la funcionalidad de históricos de eventos fueron verificados los 20 tipos de eventos mediante la herramienta de reporte personalizado creado en la interfaz gráfica del usuario.

La rapidez de acceso y uso del sistema se optimizaron mediante depuraciones de códigos del firmware.

Con respecto a la carga de datos se evidenció que en ciertas ocasiones no permitía cambiar valores en la base de datos debido al mecanismo de bloque del fichero del SQLite, pero sin revestir problemas en el desempeño del dispositivo.

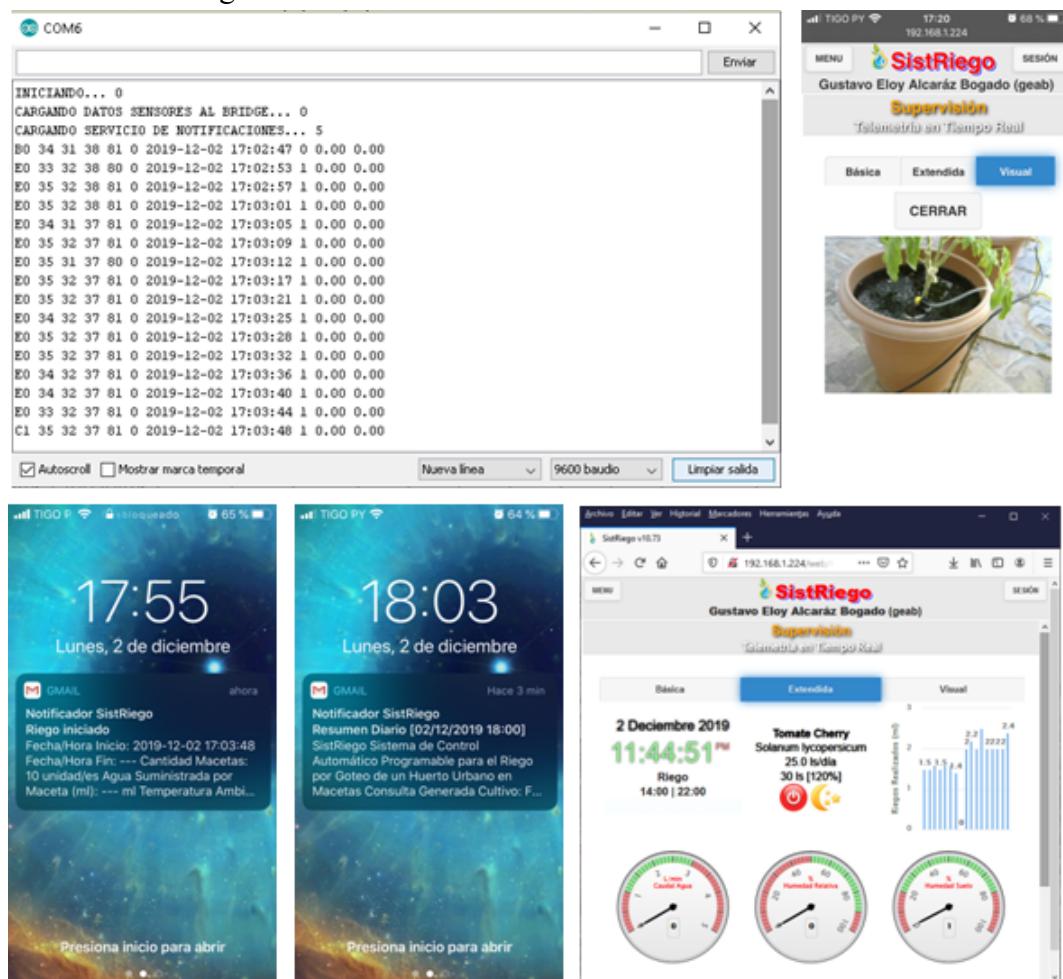
Por último la interfaz gráfica del usuario ayudó bastante para detectar inconsistencias de datos ejecutados por los programas externos del firmware, con ello se depuró y optimizó los códigos de los distintos tipos de lenguajes de programación utilizados.

A continuación se muestran fotografías captadas en los días de pruebas, algunas capturas en fecha 02/12/2019 como ejemplo de funcionamiento.

Figura 82: Ambiente de prueba. Fuente: Autor



Figura 83: Datos obtenidos 02/12/2019. Fuente: Autor



Redacción del informe final. Se redactó el informe final en base a las normas APA (American Psychological Association) en su sexta edición.

En la redacción del informe final se utilizó una herramienta informática denominada “LyX” que es un procesador avanzado de documentos de código abierto que funciona en Linux/Unix, Windows y Mac OS X, se llama “procesador de documentos” porque, a diferencia de los procesadores de texto habituales, LyX fomenta para la escritura un enfoque basado en la estructura del documento, no en su apariencia. LyX permite concentrarse en la escritura, dejando para el software los detalles del diseño visual. LyX automatiza el formato de acuerdo con un conjunto de reglas predefinidas, brindando consistencia completa incluso en los más complejos documentos. LyX genera salida de alta calidad, profesional -usando en segundo plano LaTeX, un motor de potencia industrial para tipografía, de código abierto.

En cuanto a recopilación de la literatura utilizada se utilizó la herramienta informática denominada “JabRef” que es un software de gestión bibliográfica que utiliza BibTeX como formato nativo. JabRef proporciona una interfaz fácil de usar para la edición de archivos de tipo BibTeX, para la importación de datos de bases de datos científicos en línea, y para la gestión y la búsqueda de archivos BibTeX, ésta herramienta es un complemento para referenciar las bibliografías en LyX.

Figura 84: LyX. Fuente: Autor

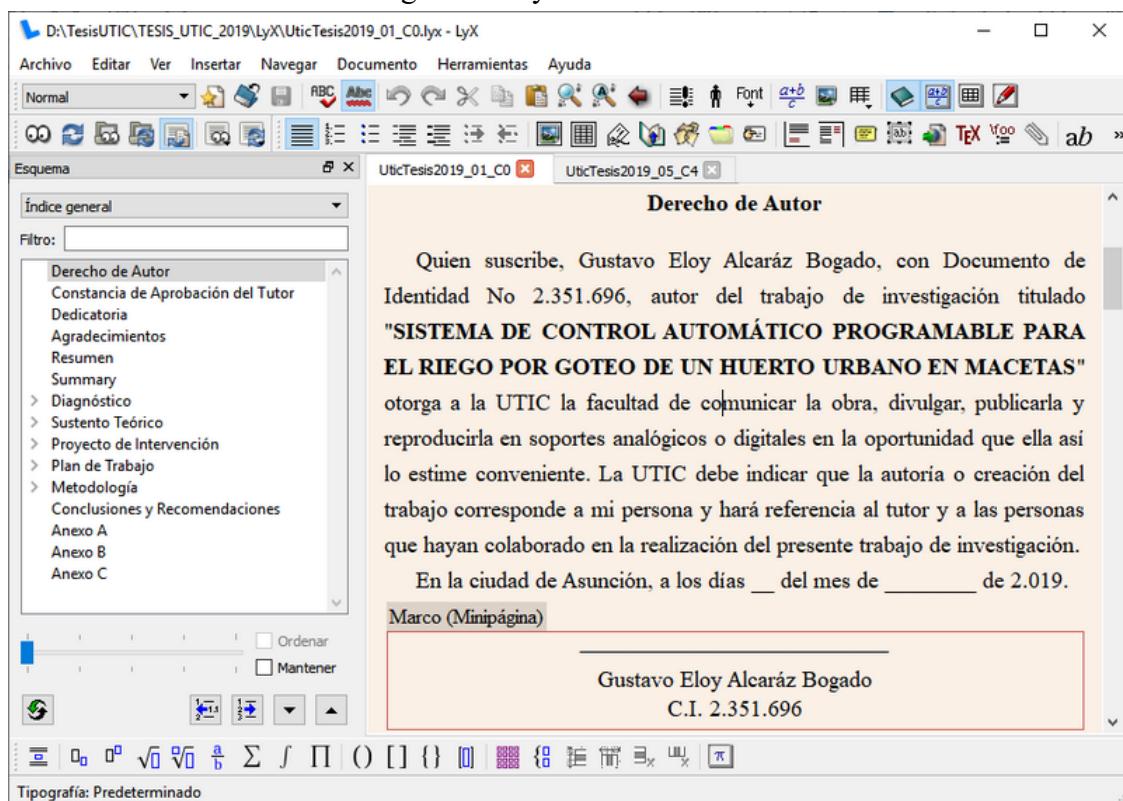


Figura 85: JabRef. Fuente: Autor

The screenshot shows the JabRef application window. The title bar reads "JabRef - D:\TesisUTIC\TESIS_UTIC_2019\LyX\bibliografia\bibliografiaUticTesis.bib (modo biblatex)". The menu bar includes Archivo, Editar, Buscar, Grupos, Ver, BibTeX, Calidad, Herramientas, Opciones, and Ayuda. Below the menu is a toolbar with various icons for file operations like Open, Save, Print, and search. A search bar with the placeholder "Buscar..." is followed by a "Filtro" button. The main area is a table titled "bibliografiaUticTesis.bib" with columns: #, author/editor, title, year, journal, bi..., and ranking. The table lists 14 entries, mostly from 2018 and 2019, related to Arduino, Raspberry Pi, and various electronic components. The last entry is a note about a local electronics store. At the bottom left, it says "Estado: JabRef está actualizado."

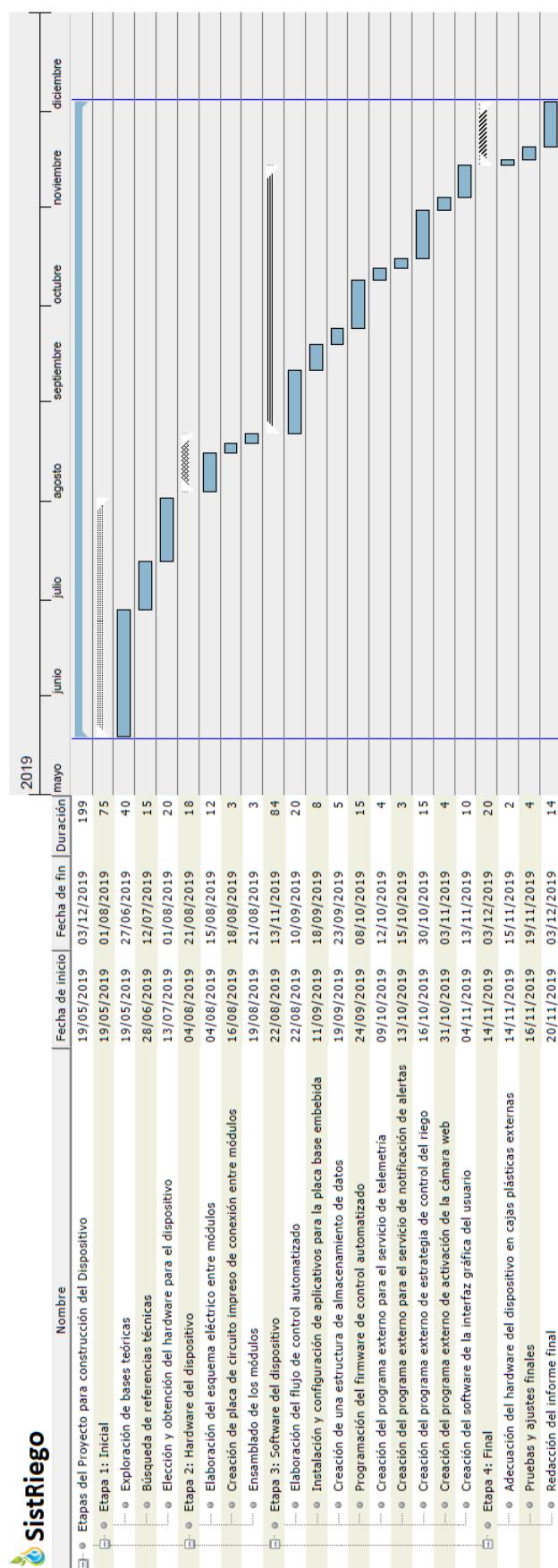
#	author/editor	title	year	journal	bi...	ranking
1	... Arriarán	Todo sobre sistemas embebidos: Arquitectura, programación y diseño de aplicaci...	2015	Arri...		
2	... Stoveken	(1 de julio de 2019) mjpg-streamer	2018	Sto...		
3	... Arlandiz	(1 de julio de 2019) Proyecto/Trabajo final de carrera: Implementación de una apli...	2009	Arl...		
4	... Plastica	(10 de junio de 2019) Práctica 2 - BreadBoard, Resistencias, LEDs y pulsadores: ...	2018	Pla...		
5	... Arduino	(11 de julio de 2019) Arduino (IDE)	2019	Ard...		
6	... Pi	(12 de julio de 2019) Raspberry Pi	2019	Pi2...		
7	... BeagreBo...	(13 de julio de 2019) Software y Hardware de código abierto para sistemas embe...	2018	Be...		
8	... Didactic	(14 de julio de 2019) Distribuidor oficial de Picaxe Educación, Arduino, Pololu, LE...	2019	Did...		
9	... Iberorobotic	(14 de julio de 2019) Electrónica y componentes	2019	Ibe...		
10	... Embajador...	(14 de julio de 2019) Productos electrónicos	2018	Em...		
11	... Mechatroni...	(14 de julio de 2019) Robótica y electrónica	2019	Me...		
12	... Electrónica	(15 de julio de 2019) Distribución de productos electrónicos	2014	Ele...		
13	... Electronics	(15 de julio de 2019) Distribuidor de semiconductores y componentes electrónicos	2019	Ele...		
**	... Gadito	(15 de julio de 2019) Vendedores de productos electrónicos en Colombia	2019	Gad...		

Recursos utilizados por Etapa

Tabla 34: Recursos utilizados en las Etapas. Fuente: Autor

Recurso Utilizado	Origen	Monto Gs.
Unidad de control (Arduino Yún)	Adquirido	500.000
Sensor de temperatura y humedad ambiente (DHT22)	Adquirido	50.000
Sensor de humedad del suelo (YL-69/YL-38)	Adquirido	35.000
Sensor de iluminación (CdS LDR GL5528)	Adquirido	2.000
Sensor de caudal (YF-S201)	Adquirido	80.000
Sensor de detección de lluvia (MH-RD/YL-38)	Adquirido	35.000
Pre-actuador (relé de un canal)	Adquirido	40.000
Actuador (Electroválvula de 12V)	Adquirido	60.000
Reloj de tiempo real (RTC DS1307)	Adquirido	35.000
Indicadores luminosos (Led RGB, Led Rojo y Led Verde)	Adquirido	3.000
Tarjetas de almacenamiento (microSD 16GB)	Adquirido	60.000
Cámara web (Logitech C270)	Adquirido	150.000
Sub-Total Guaraníes:		1.050.000

Diagrama de Gantt Ejecutado



Metodología

Tipo de Investigación

El tipo de investigación es tecnológica. La investigación tecnológica constituye un conocimiento aplicado y de uso práctico de manera inmediata, concretada en inventos, diseños, innovaciones, generalmente todos negociables para el sector productivo, en ámbitos de extracción y transformación, como los relativos a la prestación de servicios como lo son las universidades, centros de investigación, sectores turísticos, hospitales, entidades financieras, industrias de ingeniería, arquitectura, entre otros. (Soto, 2011).

Modalidad

Este trabajo, toma la modalidad de Investigación Adaptativa. La investigación adaptativa es la adaptación de conocimiento o tecnología ya existente para ser adoptada por los beneficiarios finales. (Málaga *et al.*, 2008).

Conclusiones y Recomendaciones

El uso y conservación del agua es responsabilidad de todos, y asumiendo ese compromiso, mejorar la gestión del agua, mediante un control automático programable del proceso de riego por goteo, en un huerto urbano en macetas, fue el propósito de este trabajo, porque también es importante preservar espacios verdes porque fomenta el respeto al medio ambiente.

Por tanto considerando lo expuesto y teniendo en cuenta los objetivos formulados en este trabajo se concluye cuanto sigue:

En relación al primer objetivo específico “Ensamblar las partes necesarias del hardware del dispositivo” se hizo realidad porque el dispositivo se encuentra íntegramente ensamblado y con todas las funcionalidades propuestas, de acuerdo al “Plan de Trabajo” concerniente a la Etapa 1 sobre la exploración de bases teóricas, la búsqueda de referencias técnicas, la elección y obtención del hardware para el dispositivo, fueron determinantes en la Etapa 2 para hardware del dispositivo, porque mediante la elaboración del esquema eléctrico entre módulos, la creación de placa de circuito impreso de conexión entre módulos y ensamblado de los módulos, fueron posibles por los conocimientos teóricos adquiridos en la Etapa 1.

En cuanto al segundo objetivo específico “Desarrollar, instalar y configurar el software para el dispositivo” se alcanzó en forma óptima debido al “Plan de Trabajo” en su Etapa 3, porque mediante la elaboración del flujo de control automatizado, la instalación y configuración de aplicativos para la placa base embebida, la creación de una estructura de almacenamiento de datos, la programación del firmware de control automatizado, la creación del programa externo para el servicio de telemetría, la creación del programa externo para el servicio de notificación de alertas, la creación del programa externo de estrategia de control de riego, la creación del programa externo de activación de la cámara web y la creación del software de interfaz gráfica del usuario, fueron contundentes para el desarrollo escalonado del software del dispositivo.

En atención al tercer y último objetivo específico “Acceder en forma remota para configurar y supervisar al dispositivo” se logró plenamente el acceso remoto en forma inalámbrica para configurar y supervisar al dispositivo, además de observar en tiempo real mediante una cámara web el ambiente de intervención, todo ésto fue posible gracias a la utilización de diferentes lenguajes de programación para cada funcionalidad específica del dispositivo, y la misma se comprobó en la Etapa 4 del “Plan de Trabajo”, luego de adecuar en cajas plásticas externas el hardware del dispositivo, para realizar las pruebas y ajustes finales.

Se puede asegurar que por medio de la obtención del objetivo general “Construir un dispositivo con indicadores de lluvia, temperatura ambiente, humedad del suelo y luz solar

para configurar y supervisar en forma remota la automatización del proceso de riego por goteo de un huerto urbano en macetas” a través de los objetivos específicos de la investigación nombrados anteriormente, se alcanzó a desplegar una respuesta eficaz en la automatización de la gestión del suministro del vital líquido: el agua, atendiendo las condiciones del medio ambiente y al tipo de cultivo.

Por último, en referencia a las recomendaciones, se siguiere ampliar este trabajo para investigar la manera de incorporar una inteligencia artificial y de ésta manera dotar al dispositivo de autonomía para auto configurarse continuamente, ya que posee información recolectada y almacenada en una base de datos por día de los riegos realizados al cultivo con sus correspondientes indicadores ambientales, y de ésta manera obtener eficiencia tanto en el suministro de agua y el desarrollo del cultivo, como también de extender sus funcionalidades hacia otros tipos de zonas verdes.

REFERENCIAS

- Achour, M., Betz, F., Dovgal, A., Lopes, N., Olson, P., Richter, G., Seguy, D., y Vrana, J. (2019). *Manual de PHP*. Recuperado de <https://www.php.net/manual/es/getting-started.php>. 69
- Alba, T. y Labajos, L. (2017). *Taller de riego y agua en huertos urbanos: Guía de Aprendizaje*. Ciudad de Madrid, España: Ciudad Huerto. 15, 16
- Alloza, J. M. (2012). *Montaje de componentes y periféricos microinformáticos (UF0465)*. Málaga, España: IC Editorial. 42
- Arduino (2019). (*11 de julio de 2019*) *Arduino (IDE)*. Recuperado de <https://www.arduino.cc/>. XVI, 70, 104, 105, 106, 107
- Arlandiz, D. H. (2009). (*1 de julio de 2019*) *Proyecto/Trabajo final de carrera: Implementación de una aplicación de video-on-demand sobre redes de fibra óptica*. Recuperado de https://upcommons.upc.edu/bitstream/handle/2099.1/8771/escrito_pfc.pdf?sequence=1&isAllowed=y. 97, 98
- Arriarán, S. S. (2015). *Todo sobre sistemas embebidos: Arquitectura, programación y diseño de aplicaciones prácticas con el PIC18F*. Universidad Peruana de Ciencias Aplicadas. XV, 101, 102
- Artero, O. T. (2013). *Arduino: curso práctico de formación*. 1a. ed. México D.F.: Alfaomega. XV, 57, 70
- Association, S. (2019). (*16 de julio de 2019*) *Standar SD*. Recuperado de <https://www.sdcard.org/developers/overview/index.html>. 125
- BeagleBoard.org, F. (2018). (*13 de julio de 2019*) *Software y Hardware de código abierto para sistemas embebidos*. Recuperado de <http://beagleboard.org/>. XVI, 108, 109
- Caldas, E. (2013). (*15 de julio de 2019*) *Vendedor de productos electrónicos en Colombia*. Recuperado de <https://www.electronicoscaldas.com/es/>. XVI, 118
- Consortium, S. (2019). *SQLite documentación oficial*. Recuperado de <https://www.sqlite.org/about.html>. 74
- de la Vega, A. (2011). *Manual huertos sostenibles en casa. Alicante Natura*. Ciudad de Alicante, España: Área de Medio Ambiente - Diputación de Alicante. 12, 13
- del Prado, A. y Lamas, N. (2013). *Fundamentos de Informática: Apuntes de Catedra*. San Fernando del Valle de Catamarca, Argentina: Editorial Científica Universitaria. 59, 61, 62, 64, 65, 66, 71, 77, 78, 81
- Didactic, I. (2019). (*14 de julio de 2019*) *Distribuidor oficial de Picaxe Educación, Arduino, Pololu, LEGO, etc.* recuperado de <http://shop.innovadidactic.com/index.php>. XVI,

117

- Docs, M. W. (2019). *JavaScript (JS)*. Recuperado de <https://developer.mozilla.org/es/docs/Web/JavaScript>. 70
- Dordogne, J. (2015). *Redes informáticas: Nociones fundamentales*. 5ta. ed. Barcelona, España: Eni Ediciones. XV, XVIII, 78, 80, 82, 83, 84, 87, 92
- Ebel, F., Idler, S., Prede, G., y Scholz, D. (2007). *Fundamentos de la técnica de automatización: Libro técnico*. Denkendorf, Alemania: Festo Didactic GmbH & Co. KG. XV, 17, 18, 35, 36, 37, 38, 39, 40, 41, 42
- EcuRed (2019). (2 de julio de 2019) *Cámara web*. Recuperado de https://www.ecured.cu/index.php?title=Cámaras_Web&oldid=3417534. 100, 101
- Electronics, M. (2019). (15 de julio de 2019) *Distribuidor de semiconductores y componentes electrónicos*. Recuperado de <https://www.mouser.es/>. XVI, 116, 117, 122, 123, 124, 125, 126, 128
- Electrónica, C. (2014). (15 de julio de 2019) *Distribución de productos electrónicos*. Recuperado de <https://www.carrod.mx>. XVI, 118
- Embajadores, E. (2018). (14 de julio de 2019) *Productos electrónicos*. Recuperado de <https://www.electronicaembajadores.com/es>. XVI, 112
- Endt, T., Brown, R., y Bursi, A. (2019). (2019, Julio 30) *Proyecto OpenWrt. Wireless Freedom*. Recuperado de <https://openwrt.org/start>. 72
- Española, L. R. A. (2003). *Diccionario de la Lengua Española*. Ciudad de México, México: Espasa Calpe Mexicana. 9
- Estefanía, C. (2017). (24 de mayo de 2019) *Sistemas de control: Educación Tecnológica / ESA*. Recuperado de <https://tecnologiaenesa.blogspot.com/2017/05/sistemas-de-control.html>. XV, 18
- Fink, D. G. y Christiansen, D. (1992). *Manual de ingeniería electrónica*. Madrid : McGraw-Hill Interamericana. 42
- Fuentes, L., Troya, J. M., y Vallecillo, A. (2001). *Lección 1: Desarrollo de Software Basado en Componentes*. Recuperado de <http://www.lcc.uma.es/av/Docencia/Doctorado/tema1.pdf>. 70
- Gardena (2013). (18 de mayo de 2019) *Programador de riego MasterControl*. Recuperado de <http://www.gardena.com/es/productos/riego/programadores-de-riego/programador-de-riego-mastercontrol/900963901>. XV, 11
- GNU/Linux (2019). (2019, Agosto 30) *Motion*. Recuperado de <https://motion-project.github.io/>. 99
- González, J. (2018). (3 de julio de 2019) *Sistemas Emprontrados vs PLC*. Recuperado de <http://whitewallenergy.com/la/blog/sistemas-empotrados-vs-plc/>. 101
- Guillén, I. V. (2011). *Introducción a los sistemas empotrados*. Barcelona, España: Universidad Abierta de Cataluña. XV, 102, 104
- Hernández, R., Fernández, C., y Baptista, P. (2010). *Metodología de la investigación*. 5a. ed.

- Ciudad de México, México: McGraw-Hill Interamericana. 4
- Iberorobotic (2019). (*14 de julio de 2019*) *Electrónica y componentes*. Recuperado de <https://www.iberobotics.com/>. XVI, 112
- K-Rain (2015). (*18 de mayo de 2019*) *Pro Ex 2.0 Controlador Habilitado via WiFi*. Recuperado de <https://www.krain.com/es-es/pro-ex-2-0-series-wifi-enabled-irrigation-controller>. XV, 11
- Kurose, J. F. y Ross, K. W. (2010). *Redes de Computadoras: Un enfoque descendente*. 5ta. ed. Madrid, España: Pearson Educación, S. A. 77, 78, 90, 92
- Logitech (2019). (*16 de julio de 2019*) *C270*. Recuperado de <https://www.logitech.com/es-es/product/hd-webcam-c270?crid=34>. XVI, 127
- Luaces, A. (2018). (*20 de mayo de 2019*) *La huerta urbana en envases*. Recuperado de <https://dateunavuelta.uy/blog/index.php?la-huerta-urbana-en-envases>. 13, 14
- Malvino, A. P. y Bates, D. J. (2007). *Principios de electrónica*. 7a ed. Madrid, España : McGraw-Hill. 43
- Marqués, M. (2011). *Universitat Jaume I.: Bases de datos (libro digital)*. Recuperado de <https://openlibra.com/es/book/download/bases-de-datos-2>. 72, 73
- Marzal, A. y Gracia, I. (2010). *Universitat Jaume I.: Introducción a la programación con C (libro digital)*. Recuperado de <https://openlibra.com/es/book/download/introduccion-a-la-programacion-con-c-3>. 67
- Mas, J. L. V. (2011a). *Fundamentos Informáticos en la Ingeniería de Edificación. Tema 1: Hardware*. Recuperado de https://rua.ua.es/dspace/bitstream/10045/19764/1/01-1_HARDWARE.pdf. 60, 61, 62
- Mas, J. L. V. (2011b). *Fundamentos Informáticos en la Ingeniería de Edificación. Tema 2: Software*. Recuperado de https://rua.ua.es/dspace/bitstream/10045/19765/1/01-2_SOFTWARE.pdf. 65, 66
- Mechatronics, N. (2019). (*14 de julio de 2019*) *Robótica y electrónica*. Recuperado de <https://naylampmechatronics.com/>. XVI, 113, 114, 115, 116, 117, 119, 120, 121, 122, 123, 128
- Mendiburu, H. (2003). *Automatización Medioambiental. Aplicación de la automatización industrial y el control de procesos, en la protección y conservación del medio ambiente*. Ciudad de Lima, Perú: Indecopi. XV, 8, 18, 19, 21, 22, 23, 24, 28, 30, 31, 32, 34
- Microsoft (2019). (*16 de julio de 2019*) *LifeCam Studio*. Recuperado de <https://www.microsoft.com/accessories/es-es/products/webcams/lifecam-studio/q2f-00009>. XVI, 127
- Miguel, P. A. S. (2014). *Electrónica: Instalaciones eléctricas y automáticas: [Electricidad - Electrónica]*. 2ª ed. Madrid, España: Paraninfo. XVIII, 44, 45, 46, 47, 48, 49
- Montiel, D. P. (2008). *Introducción a SQLite*. Alicante, España: I.E.S. San Vicente. 75, 76
- Munch, L. y Ángeles, E. (1988). *Métodos y técnicas de investigación para administración e ingeniería*. México: Trillas. 6
- Málaga, J. T., Vera, G., y Ramnos, R. O. (2008). (*30 de agosto de 2019*) *Tipos, métodos y estrategias de investigación científica*. Recuperado de <http://www.imarpe.gob.pe/>

- imarpe/ archivos/ articulos/ imarpe/ oceanografia/ adj_modela_pa- 5- 145- tam- 2008-
investig. pdf. 193
- Nelson (2011). (*18 de mayo de 2019*) *Single Outlet Electronic Timer: 56607 Water Timer.* Recuperado de <http://www.lrnelson.com/products/water-conservation2/water-timers/single-outlet-electronic-timer>. XV, 10
- of Art, W. S. (2012). (*31 de junio de 2019*) *Time-Based Media: Video Basics.* Recuperado de <http://wsa.wikidot.com/tbm:video-basics>. 94, 95, 96
- Ogata, K. (2010). *Ingeniería de control moderna.* 5ta. ed. Madrid, España: Prentice Hall. 20, 21
- Ojeda, L. R. (2015). *Escuela Superior Politécnica del Litoral: Python Programación (libro digital).* Recuperado de <https://openlibra.com/es/book/download/python-programacion>. 68
- Olifer, N. y Olifer, V. (2009). *Redes de Computadoras.* España: McGraw-Hill Interamericana. XVIII, 77, 78, 88, 89, 90, 91, 92
- Oncina, S. y Castelló, M. (2013). *Huertos ecológicos en balcones y terrazas.* Alicante, España: Red de Huertos urbanos de Alicante. 14
- Ossa, E. S. P. y Ballesteros, J. C. C. (2012). *GNU/Linux Sistema Operativo y Servicios: Configuración Básica de Servicios en GNU/Linux.* Bogotá, Colombia: SENA. 91, 92, 94
- Pi, R. (2019). (*12 de julio de 2019*) *Raspberry Pi.* Recuperado de <https://www.raspberrypi.org/>. XVI, 107, 108
- Plastica (2018). (*10 de junio de 2019*) *Práctica 2 - BreadBoard, Resistencias, LEDs y pulsadores: Conceptos Teóricos.* Recuperado de <http://plastika.es/practica-2-breadboard-resistencias-leds-y-pulsadores>. XV, 59
- Razer (2019). (*16 de julio de 2019*) *Kiyo.* Recuperado de <https://www2.razer.com/es-es/gaming-broadcaster/razer-kiyo>. XVI, 127
- República del Paraguay, P. L. (1996). *Ley que sanciona delitos contra el medio ambiente N° 716/96.* Congreso de la Nación (1996). XI, 131
- República del Paraguay, P. L. (2007). *Ley de los Recursos Hídricos del Paraguay N° 3239/2007.* Congreso de la Nación (2007). XI, 129
- Schilling, D. L. y Charles, B. (1991). *Circuitos electrónicos discretos e integrados.* Barcelona, España: Alfaomega - Marcombo. XVIII, 55, 56
- Schonwald, J. y Pescio, F. (2015). *Mi casa, mi huerta. Técnicas de agricultura urbana: Colección Divulgación.* Buenos Aires, Argentina: Inta. XVIII, 13, 14
- Soto, I. B. R. (2011). (*28 de agosto de 2019*) *I.B.: Investigación científica e Investigación tecnológica como componentes para la Innovación: Consideraciones técnicas y metodológicas, en Contribuciones a las Ciencias Sociales.* Recuperado de <http://www.eumed.net/rev/cccss/13/ibrs.html>. 193
- Stovelen, T. (2018). (*1 de julio de 2019*) *mjpg-streamer.* Recuperado de <https://github.com/jacksonliam/mjpg-streamer>. 99

- Talca, C. E. S. (2014). (*6 de junio 2019*) *Medición y Análisis de Componentes y Circuitos Electrónicos (libro digital)*. Recuperado de <https://www.academia.edu/8673861/>. 43, 44, 48, 51, 52, 53, 58
- Tanenbaum, A. S. y Wetherall, D. J. (2012). *Redes de computadoras*. 5ta. ed. México: Pearson Educación. XVIII, 76, 77, 78, 92
- Vassallo, F. R. (2000). *Enciclopedia del Técnico en electrónica: Componentes Electrónicos: para audio e imagen*. Perú: Ediciones CEAC. 49, 53

GLOSARIO DE TÉRMINOS

ACID	En concreto ACID es un acrónimo de Atomicity, Consistency, Isolation and Durability: Atomicidad, Consistencia, Aislamiento y Durabilidad en español, es un conjunto de características o propiedades que garantizan que las transacciones en una base de datos son fiables.
Apotema	Distancia entre el centro de un polígono regular y uno cualquiera de sus lados.
Bare-Metal	Los servidores bare metal proporcionan a los usuarios acceso exclusivo a todo el servidor. A diferencia de un servidor virtual con múltiples arrendatarios, el servidor bare metal puro es de un solo arrendatario y se ofrece sin hipervisor, eliminando así el efecto ""vecino ruidoso"", así como cualquier ""impuesto"" de rendimiento del hipervisor.
Caudalímetro	Instrumento para la medición de caudal o gasto volumétrico de un fluido.
Circuito	Conjunto de conductores que recorre una corriente eléctrica, y en el cual hay generalmente intercalados aparatos productores o consumidores de esta corriente.
dispositivo	Pieza o conjunto de piezas o elementos preparados para realizar una función determinada y que generalmente forman parte de un conjunto más complejo.
DSP	Un procesador de señal digital (DSP) es un tipo de microprocesador (increíblemente rápido y poderoso).
electroválvula	Válvula accionada por un electroimán, que regula un circuito hidráulico o neumático.
eMMC	(embedded MultiMediaCard) Tipo de memoria de corta capacidad y reducido tamaño y precio, su consumo también es bastante reducido, este sirve principalmente para dispositivos de reducido tamaño, como por ejemplo un teléfono móvil o una tableta e incluso como para cámaras.
estocástico	Perteneciente o relativo al azar, comportamiento aleatorio.
fitosanitario	Perteneciente o relativo a la prevención y curación de las enfermedades de las plantas.

Fundamento	Principio y cimiento en que estriba y sobre el que se apoya un edificio u otra cosa, razón principal o motivo con que se pretende afianzar y asegurar algo
Hardware	Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático (tangible y/o visible)
homologación	Derivado del griego homologos ""acordar"", es el término que se usa en varios campos para describir la equiparación de las cosas, ya sean éstas características, especificaciones o documentos.
ICSP	(In Circuit Serial Programming) sirve para programar el BootLoader del Microcontrolador ATmega y así poder cargar los programas que creemos en el Arduino IDE directamente en el microcontrolador sin tener que necesitar Programadores externos.
IEEE	El Instituto de Ingeniería Eléctrica y Electrónica es una asociación mundial de ingenieros dedicada a la normalización y el desarrollo en áreas técnicas, fundada el 1 de enero de 1963, su trabajo es promover la creatividad, el desarrollo y la integración, compartir y aplicar los avances en las tecnologías de la información, electrónica y ciencias en general para beneficio de la humanidad y de los mismos profesionales.
índicador	Dato o información que sirve para conocer o valorar las características y la intensidad de un hecho o para determinar su evolución futura.
LaTeX	Es un sistema de composición de textos, orientado a la creación de documentos escritos que presenten una alta calidad tipográfica. Por sus características y posibilidades, es usado de forma especialmente intensa en la generación de artículos y libros científicos que incluyen, entre otros elementos, expresiones matemáticas.
Memoria	Dispositivo que retiene, memoriza o almacena datos informáticos durante algún período de tiempo
mnemotécnico	Es un sistema sencillo utilizado para recordar una secuencia de datos, nombres, números, y en general para recordar listas de ítems que no pueden recordarse fácilmente.
OpAmp	El Amplificador Operacional también llamado OpAmp, o Op-Amp es un circuito integrado. Su principal función es amplificar el voltaje con una entrada de tipo diferencial para tener una salida amplificada y con referencia a tierra.
organoléptica	Que produce una impresión sensorial.

PCB	Del inglés: Printed Circuit Board, PCB, es decir, placa de circuito impreso.
presión	Magnitud que expresa la fuerza que actúa sobre una unidad de superficie.
PRU	Es un tipo de microcontrolador (Programmable Real-Time Unit - unidad programable en tiempo real)
PWM	(pulse-width modulation) La modulación por ancho de pulsos de una señal o fuente de energía es una técnica en la que se modifica el ciclo de trabajo de una señal periódica, ya sea para transmitir información a través de un canal de comunicaciones o para controlar la cantidad de energía que se envía a una carga
RFC	Las RFC (Request For Comments, Peticiones de comentarios) son un conjunto de documentos que sirven de referencia para la comunidad de Internet, que describen, especifican y asisten en la implementación, estandarización y discusión de la mayoría de las normas, los estándares, las tecnologías y los protocolos relacionados con Internet y las redes en general, redactadas por una organización llamada IETF (Internet Engineering Tasking Force, Grupo de Trabajo de Ingeniería de Internet).
Software	Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas (no visible, abstracto).

Anexo A

Fotos de ubicación espacial del huerto urbano



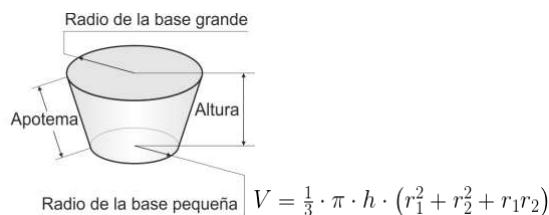
Fotos de la distribución de las macetas del huerto urbano



Fotos del sistema de riego de macetas del huerto urbano



Cálculo de volumen de un cono recto truncado



Donde:

r1= radio de la base grande = 15 cm

r2= radio de la base pequeña = 10 cm

h= altura = 30 cm

$$V = \frac{1}{3} * 3.14 * 30 * (15^2 + 10^2 + 15 * 10) = 14922,5 \text{ cm}^3 = 14,92 \text{ litros}$$

INGENIERÍA EN SISTEMAS INFORMÁTICOS

SISTEMA DE CONTROL DE AUTOMATIZACIÓN PROGRAMABLE CON ACCESO REMOTO DEL PROCESO DE RIEGO POR GOTEO DE UN HUERTO URBANO EN MACETAS

Trabajo de campo mediante entrevista estructura

DATOS DEL ENTREVISTADO/A

1

Nombres: Claudio Rufino
Profesión: Ingeniero Agrónomo

Apellidos: Cabañas Gálvez
Edad: 36 años.

C.I.P.: 1.711.025

Fecha: 18/Mayo/2019

CONSIDERACIONES A TENER EN CUENTA

Huerto urbano con un área de 3,685 metros cuadrados de espacio utilizado, cuyas dimensiones son de 1 metro de ancho por 2,5 metros de largo ubicado al costado izquierdo de la edificación en el patio lateral de mi residencia en sitio calle de la Amistad Nro. 172 de la ciudad de San Lorenzo del Departamento Central de la República del Paraguay, se compone por una distribución de 10 macetas jardineras exteriores en dos filas y separadas 40 cm entre sí

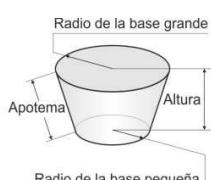


sobre un suelo revestido por pisos de tipo cerámico poroso con diseño de piedras estilo rompecabezas, las macetas jardineras están fabricadas en material plástico de color marrón oscuro en forma de cono rectotrunco invertido teniendo 4 agujeros pequeños en el fondo para el drenaje, sus dimensiones son de 35 centímetros de diámetro de base grande, 30 centímetros de apotema (altura) y 20 centímetros diámetro de base pequeña, cuentan con sus respectivos platos de retención de líquido excedente con un diámetro de 25 centímetros, con un volumen de capacidad de aproximadamente 14,92 litros, conteniendo en su interior un sustrato compuesto por una mezcla de humus de lombriz

con tierra normal y como base piedra tritura número 4, en ellas se encuentran cultivadas plantas de tomate cherry (*Solanumlycopersicum* variedad Cerasiforme) teniendo una altura promedio de 38 centímetros, el huerto urbano tiene implementado un proceso manual de riego por goteo, compuesto por 15 metros de micro tubo de 6 milímetros de diámetro exterior y 4 milímetros de diámetro interior, 10 goteros regulables con piqueta con una capacidad de suministro de agua de 6 litros/horas, 11 uniones tipo te (T), 2 codos de 90°, 2 terminadores de línea y 1 conector con reducción de boca de 3/4 pulgadas a 4 milímetros conectada en forma directa a la red de distribución de agua corriente local (ESSAP) por medio de una canilla para lavarropas con un pico de 3/4 de pulgadas, por último el área afectada está cubierta con un techo translucido en policarbonato con una altura de 2,40 metros que deja pasar en gran medida la luz solar y se encuentra separada a 1 metro de la pared de la casa teniendo un corredor de aire para su ventilación hacia la calle.



Diseño maceta:



Radio base grande: 15 cm
Apotema: 30,0 cm
Radio base pequeña: 10 cm
V= 14,92 litros

Esquema de distribución de macetas:

ANCHO	Dimensiones del área de investigación Unidad de medida metros										AREA (m2)	
	M	E	M	E	M	E	M	E	M			
1,1 m	0,35	0,40	0,35	0,40	0,35	0,40	0,35	0,40	0,35	3,685		
	0,40	0,40	0,40	0,40	0,40	0,40	0,40	0,40	0,40			
	0,35	0,40	0,35	0,40	0,35	0,40	0,35	0,40	0,35			
LARGO	3,35 m											
Referencias: M= Macetas E=Espacio entre macetas												

PREGUNTAS TENIENDO EN CUENTA LAS MACETAS

1. ¿Qué tipos de cultivos puede albergar, a parte del actual, con esas dimensiones?

INGENIERÍA EN SISTEMAS INFORMÁTICOS

**SISTEMA DE CONTROL DE AUTOMATIZACIÓN PROGRAMABLE CON ACCESO REMOTO
DEL PROCESO DE RIEGO POR GOTEO DE UN HUERTO URBANO EN MACETAS**

Trabajo de campo mediante entrevista estructura

2

- Respuesta:
2. ¿Se puede cultivar más de un cultivo, en caso de afirmación cuántos y qué tipos de cultivos son combinables?
Respuesta:
 3. ¿El sustrato de las macetas se deben cambiar por cada nuevo cultivo?
Respuesta:
 4. ¿Cuál sería el sustrato adecuado para este tipo de macetas en forma general?
Respuesta:
 5. ¿Ha realizado alguna vez este tipo de huerto urbano en macetas?
Respuesta:
 6. ¿Tiene algún aporte adicional o recomendaciones sobre este apartado?
Respuesta:

PREGUNTAS TENIENDO EN CUENTA EL RIEGO POR GOTEO

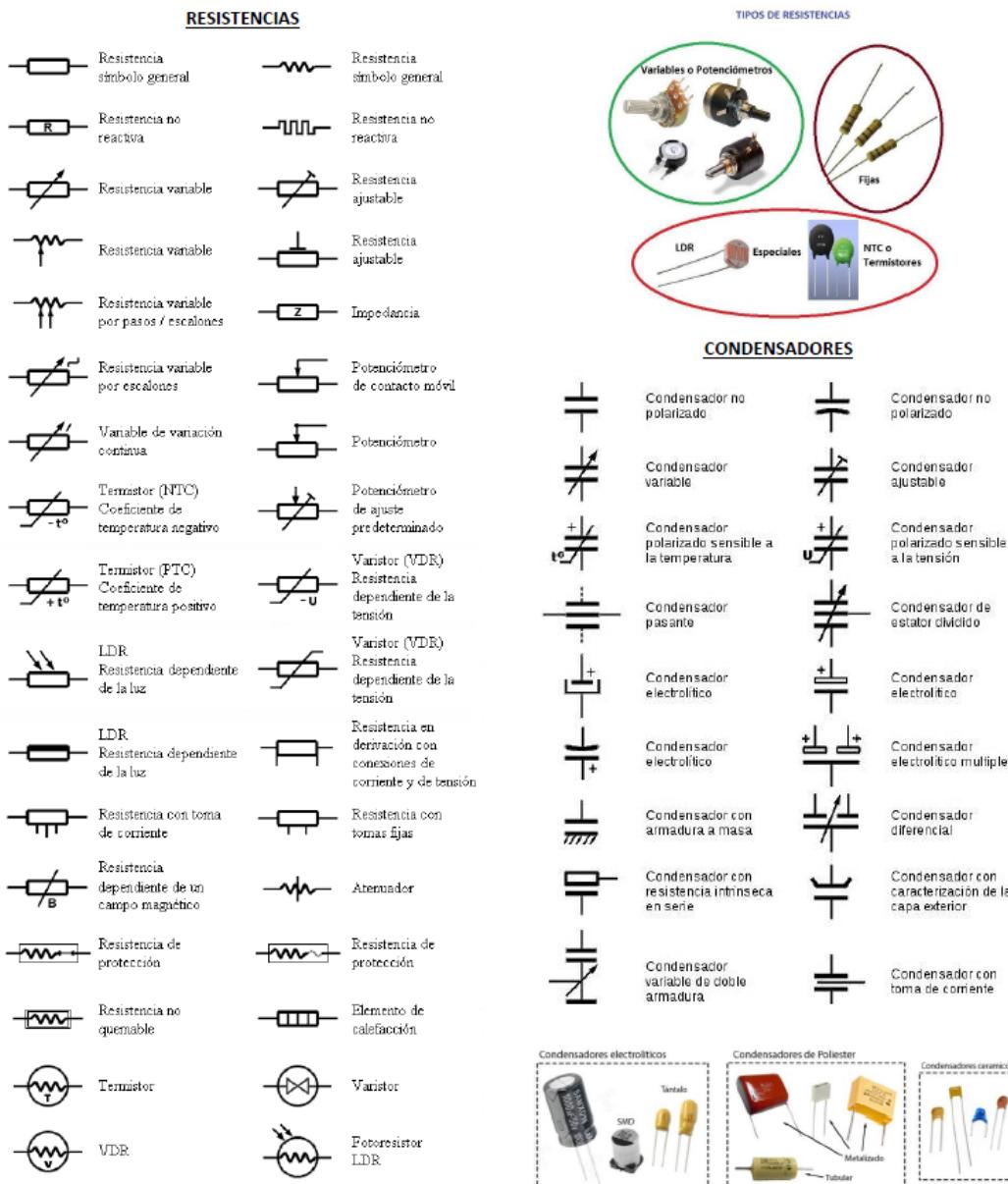
1. ¿Qué cantidad de presión de agua (bares) necesita este tipo de riego por goteo?
Respuesta:
2. ¿Qué frecuencia de riego sería el adecuado en la semana y qué duración deberían tener por cada cultivo?
Respuesta:
3. ¿En qué rango de intensidad porcentual de luz solar es recomendable realizar el riego (0-100%)?
Respuesta:
4. ¿En qué rango de temperatura de ambiente en grados Celsius aproximadamente se puede realizar el riego (0-50 °C)?
Respuesta:
5. ¿En qué porcentaje de humedad relativa del suelo es recomendable el riego (0-100 %)?
Respuesta:
6. ¿Qué cantidad de agua necesita aproximadamente cada maceta (cm³ o litros)?
Respuesta:
7. ¿Se deben tener en cuenta la evapotranspiración en estas macetas para el riego?
Respuesta:
8. ¿Qué cuidados de mantenimiento se deben tener por este tipo de riego por goteo?
Respuesta:
9. ¿Tiene algún aporte adicional o recomendaciones sobre este apartado?
Respuesta:

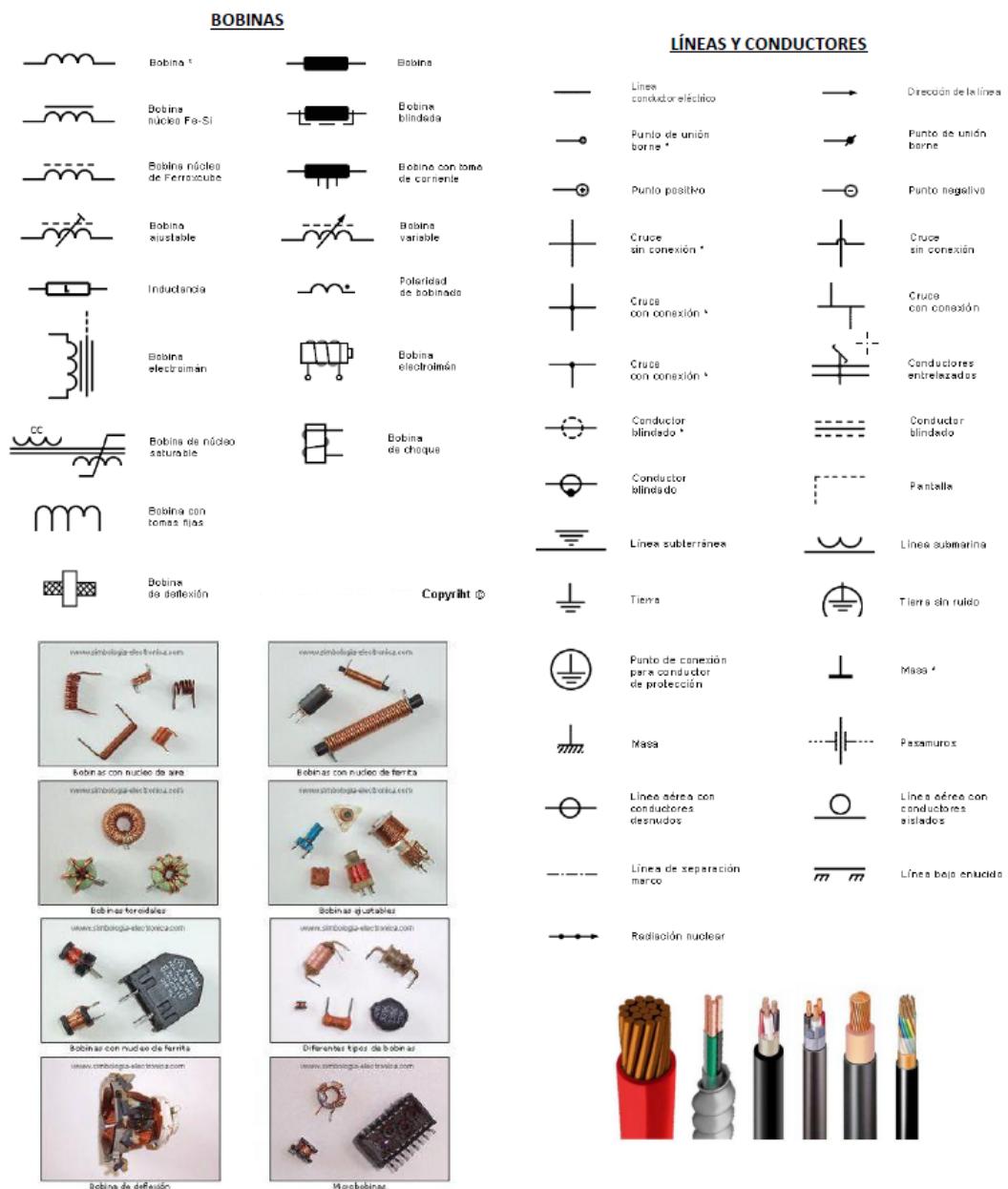
MUCHAS GRACIAS POR SU APORTE Y TIEMPO

El presente trabajo tiene como objetivo realizar un relevamiento técnico para crear un prototipo de dispositivo electrónico/informático que configure, programe (horarios/duración) y automatice el proceso de riego por goteo de un huerto urbano en macetas mediante sensores de: iluminación, temperatura, humedad y de lluvia, y por último registrar en una base de datos los procesos de riegos efectuados con las condiciones ambientales del momento, cantidad de agua suministrada y duraciones del proceso con el propósito de realizar una minería de datos mediante reportes estadísticos que el sistema debe arrojar, implementando para el efecto una interfaz gráfica de usuario en plataforma web con acceso inalámbrico al dispositivo para su configuración y monitoreo.

Anexo B

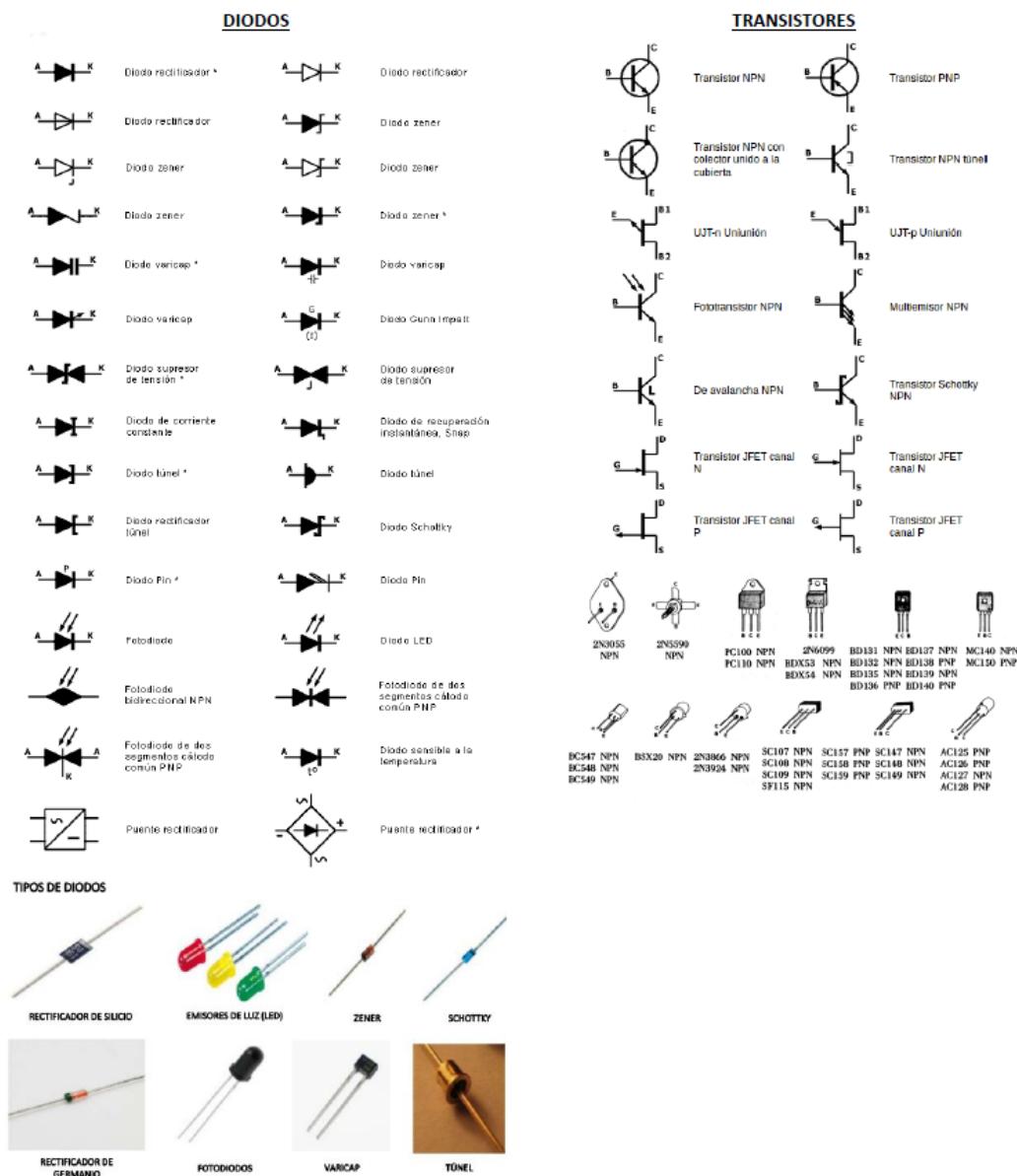
• Símbolos electrónicos



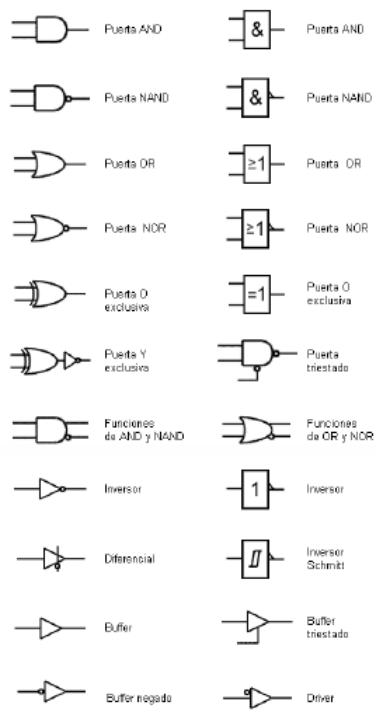


INTERRUPTORES		FUSIBLES	
	Interruptor contacto abierto		Interruptor contacto cerrado
	Pulsador contacto abierto *		Pulsador contacto cerrado *
	Pulsador contacto abierto		Pulsador contacto cerrado
	Commutador dos posiciones		Commutador multiposiciones
	Commutador multiposiciones		Fusible de operación rápida
	Interruptor doble uno cierra antes que el otro		Fusible de operación lenta
	Pulsador que actúa sobre dos circuitos		Disyuntor térmico
	Interruptor que actúa sobre dos circuitos		Delimitador para cable de alimentación
	Contacto cerrado con retraso al abrir		Contacto abierto con retraso, tanto al abrir como al cerrar
	Selector		Contacto abierto con retraso, tanto al abrir como al cerrar
	Botón pulsador		Botón pulsador con señalización lumínica
			Símbolo del fusible símbolo genérico
			Símbolo del fusible símbolo genérico
			Fusible
			Fusible
			Fusible de operación rápida
			Fusible de operación lenta
			Disyuntor térmico
			Delimitador para cable de alimentación
			Contacto con fusible
			El lado ancho es el lado de la red
			Fusible con contacto de alarma
			Indicador contacto de alarma
			Resistencia de protección
			Resistencia de protección



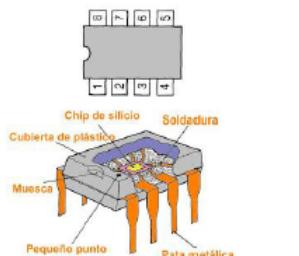


CIRCUITOS INTEGRADOS

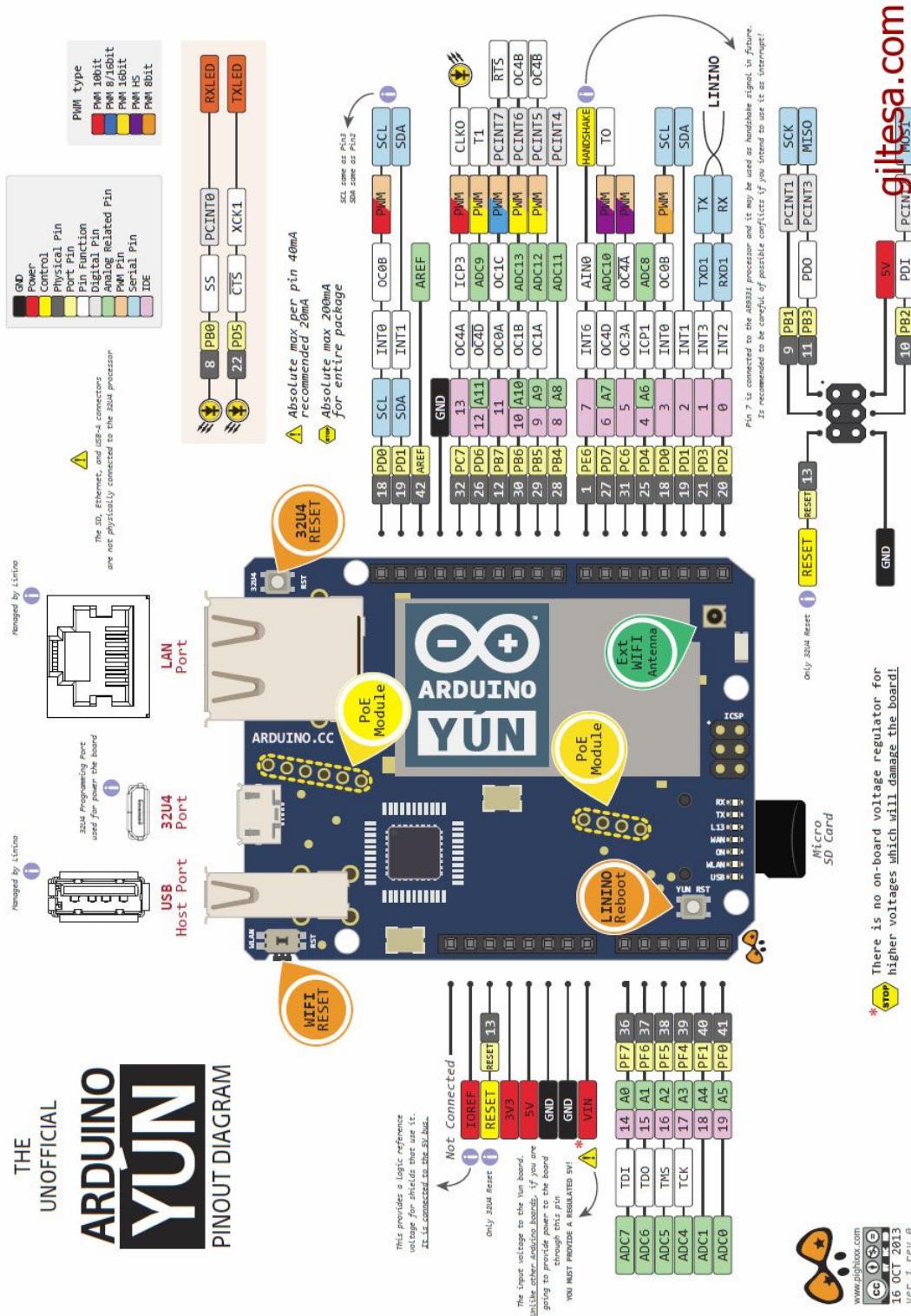


FUNCIONES LÓGICAS BÁSICAS

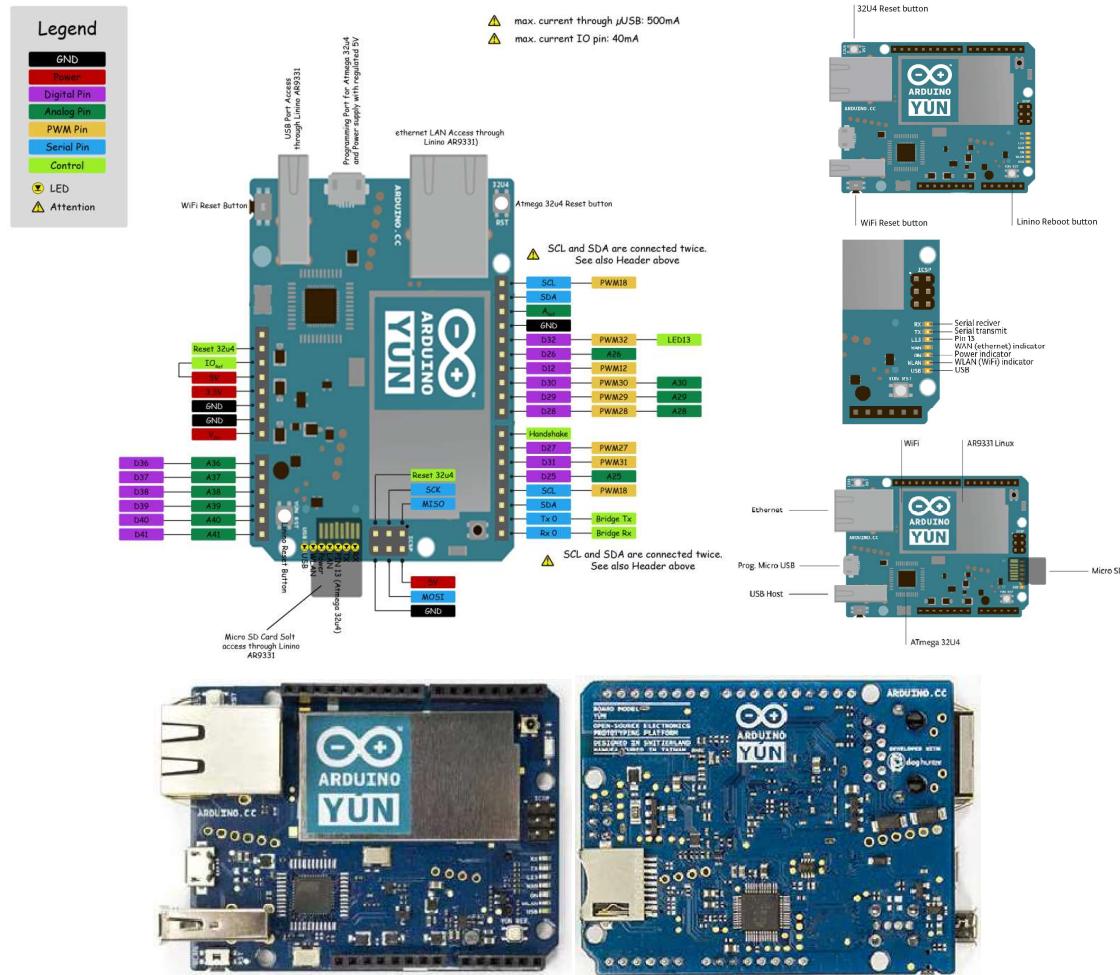
NOMBRE	AND - Y	OR - O	XOR O-exclusiva	NOT Inversor	NAND	NOR
SÍMBOLO						
SÍMBOLO						
TABLA DE VERDAD						
EQUIVALENTE EN CONTACTOS						
AXIOMA	$z = a \cdot b$	$z = a + b$	$z = \bar{a} \cdot \bar{b} + a \cdot \bar{b}$	$z = \bar{a}$	$z = \bar{a} \cdot \bar{b}$	$z = \bar{a} + \bar{b}$



Anexo C

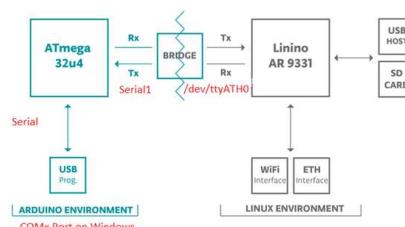


UNIDAD DE CONTROL ARDUINO YÚN



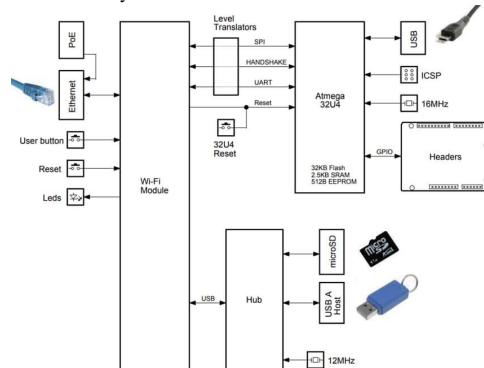
AVR Arduino Microcontrolador

Microcontroller ATmega32u4
Operating Voltage 5V
Input Voltage 5V
Digital I/O Pins 20
PWM Channels 7
Analog Input Channels 12
DC Current per I/O Pin 40 mA
DC Current for 3.3V Pin 50 mA
Flash Memory 32 KB (of which 4 KB used by bootloader)
SRAM 2.5 KB
EEPROM 1 KB
Clock Speed 16 MHz

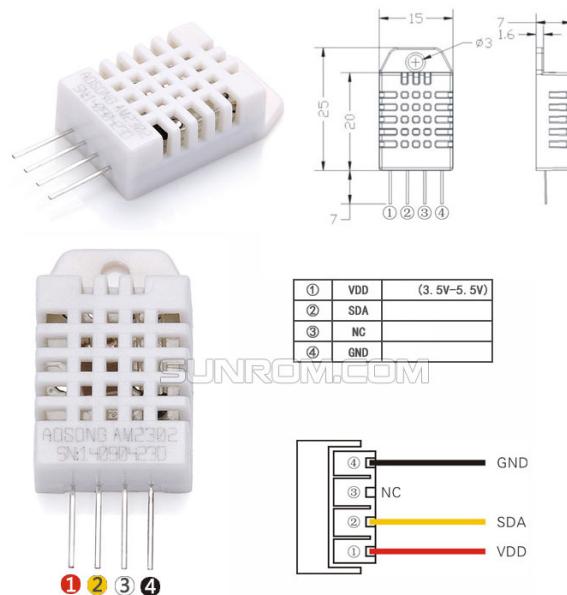


Linux Microprocesador

Processor Atheros AR9331
Architecture MIPS @400MHz
Operating Voltage 3.3V
Ethernet IEEE 802.3 10/100Mbit/s
WiFi IEEE 802.11b/g/n
USB Type-A 2.0 Host/Device
Card Reader Micro-SD only
RAM 64 MB DDR2
Flash Memory 16 MB

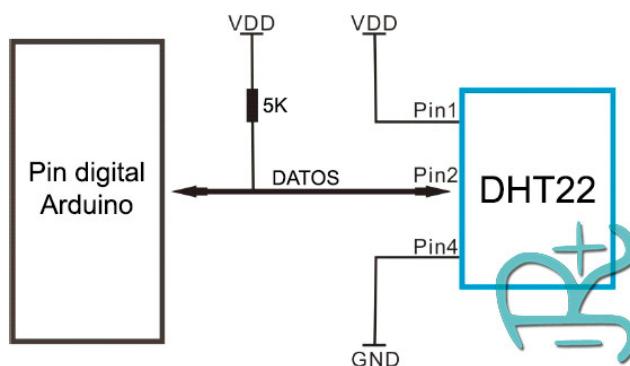


SENSOR DE TEMPERATURA Y HUMEDAD AMBIENTE DHT22

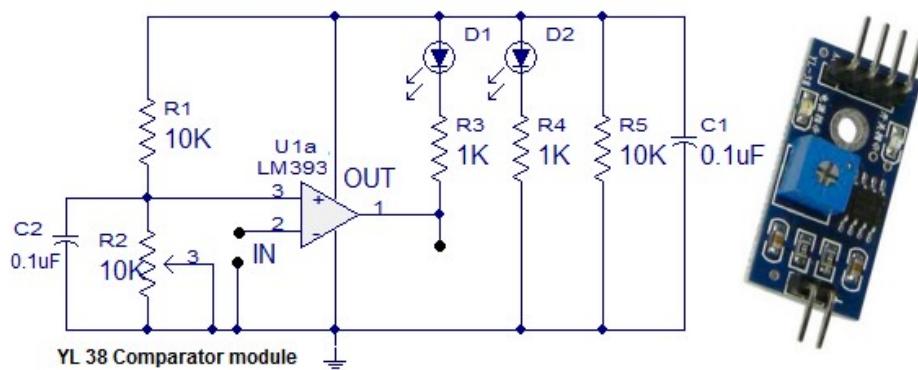
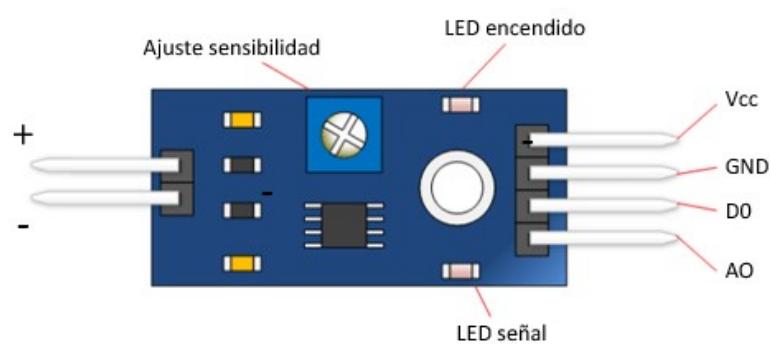
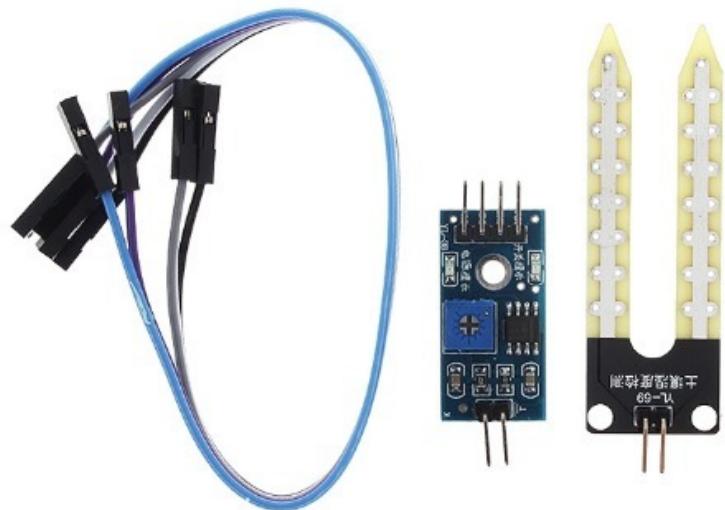


Technical Specification:

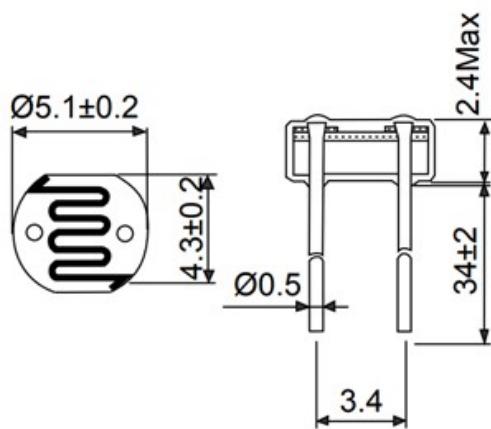
Model	DHT22
Power supply	3.3-6V DC
Output signal	digital signal via single-bus
Sensing element	Polymer capacitor
Operating range	humidity 0-100%RH; temperature -40~80Celsius
Accuracy	humidity +/-2%RH(Max +/-5%RH); temperature <+/-0.5Celsius
Resolution or sensitivity	humidity 0.1%RH; temperature 0.1Celsius
Repeatability	humidity +/-1%RH; temperature +/-0.2Celsius
Humidity hysteresis	+/-0.3%RH
Long-term Stability	+/-0.5%RH/year
Sensing period	Average: 2s
Interchangeability	fully interchangeable
Dimensions	small size 14*18*5.5mm; big size 22*28*5mm



SENSOR DE HUMEDAD DEL SUELO YL-69/YL-38

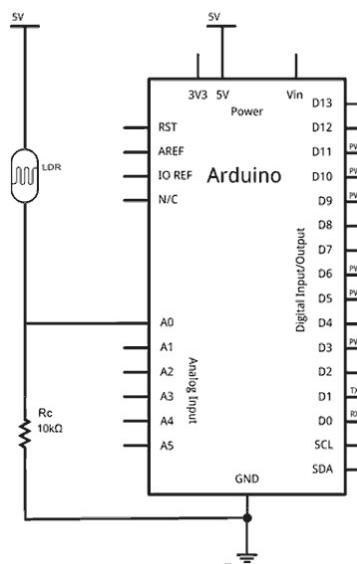


SENSOR LDR GL55 (iluminación)



Types and Specifications

Specification	Type	Max. Voltage	Max. power	Environmental temp.	Spectrum peak value
Φ5 series	GL5516	150	90	-30~+70	540
	GL5528	150	100	-30~+70	540
	GL5537-1	150	100	-30~+70	540
	GL5537-2	150	100	-30~+70	540
	GL5539	150	100	-30~+70	540
	GL5549	150	100	-30~+70	540



SENSOR DE FLUJO DE LÍQUIDO YF-S201

Features:

- Compact, Easy to Install
- High Sealing Performance
- High Quality Hall Effect Sensor
- RoHS Compliant

Specifications:

Working Voltage: DC 4.5V~24V
 Normal Voltage: DC 5V~18V
 Max Working Current: 15mA (DC 5V)
 Load capacity: ≤ 10 mA (DC 5V)
 Flow Rate Range: 1~30L/min
 Load Capacity: ≤10mA (DC 5V)
 Operating Temperature: ≤80°C
 Liquid Temperature: ≤120°C
 Operating Humidity: 35%~90%RH
 Allowing Pressure: ≤1.75MPa
 Storage Temperature: -25~+80°C
 Storage Humidity: 25%~95%RH
 Electric strength 1250V/min
 Insulation resistance ≥ 100MΩ
 External threads: 1/2"
 Outer diameter: 20mm
 Intake diameter: 9mm
 Outlet diameter: 12mm



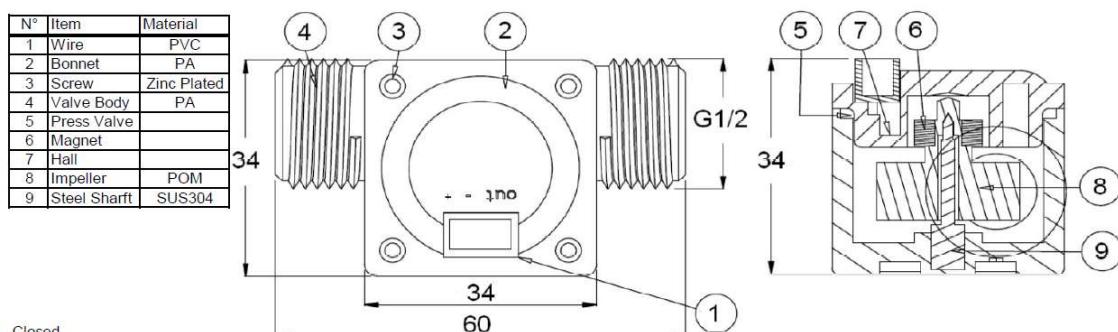
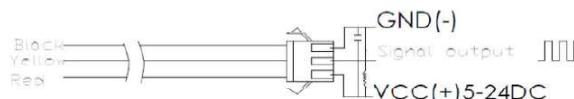
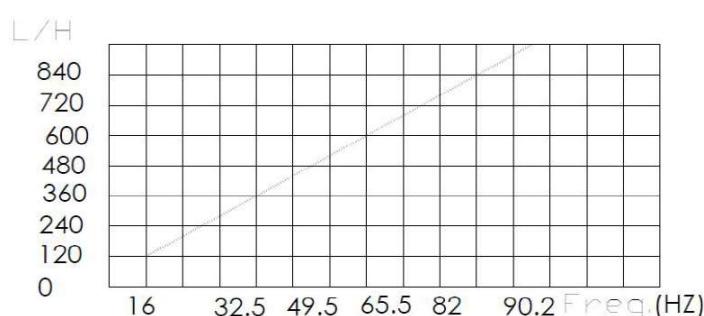
Application:

Water heaters, credit card machines, water vending machine, flow measurement device

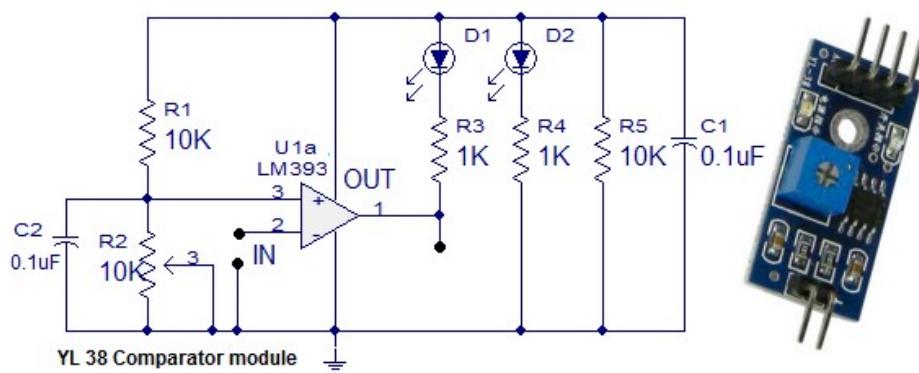
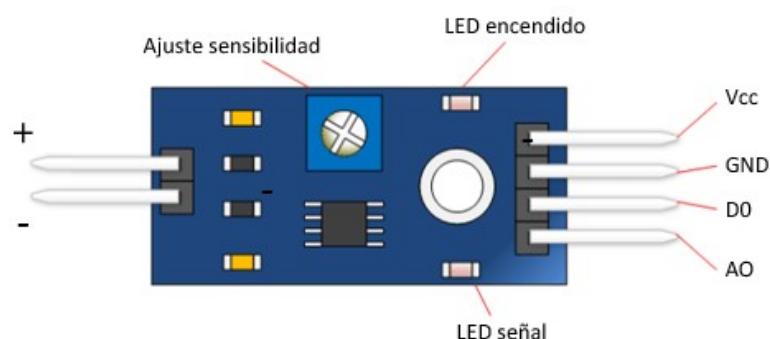
Circuit:

Red: Positive
 Black: GND
 Yellow: Output signal

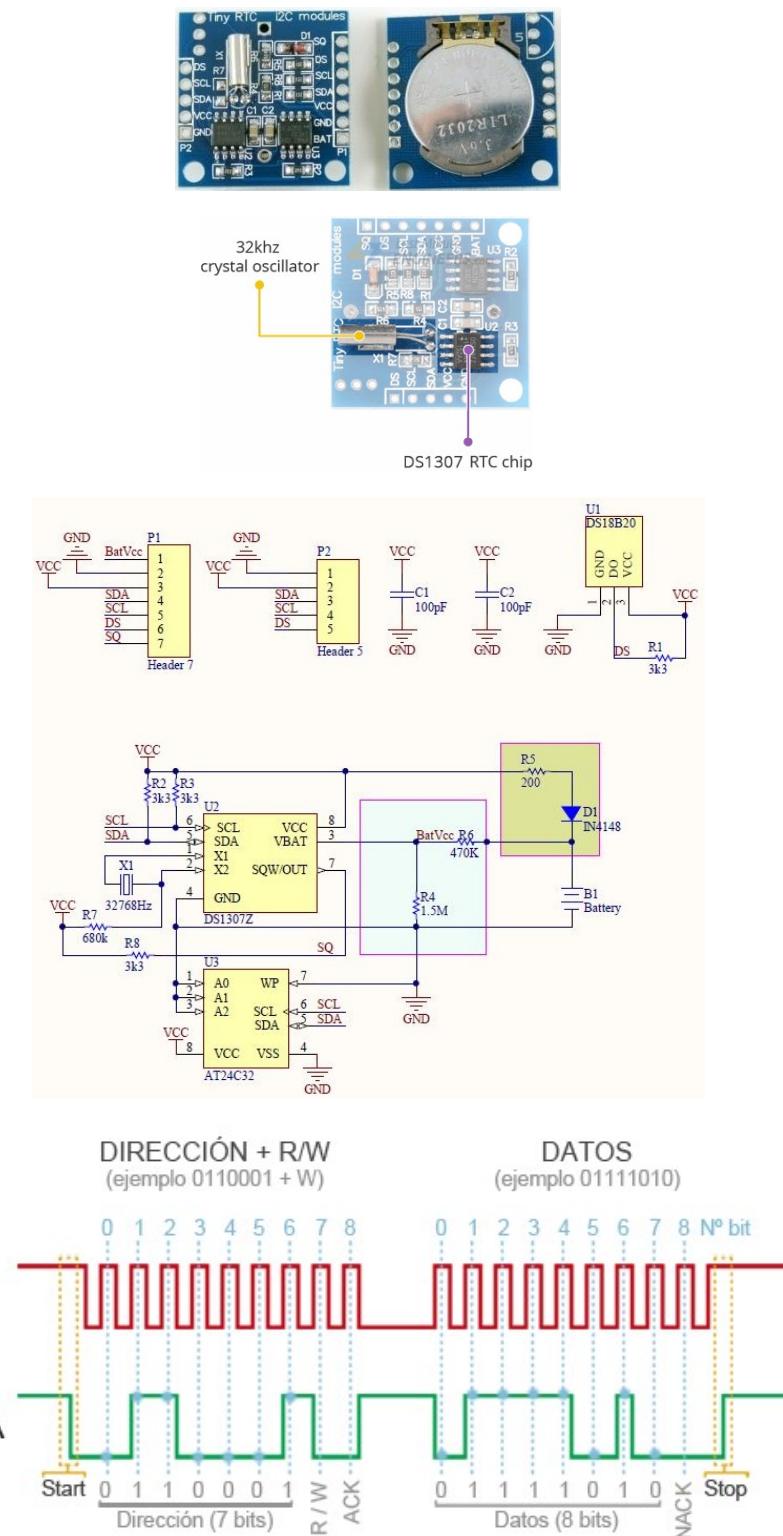
Flow Range: 100L/H-1800H-L/H		
Flow (L/H)	Freqz.(Hz)	Err range
120	16	
240	32.5	
360	49.3	±10
480	65.5	5%
600	82	
720	90.2	



SENSOR DE DETECCIÓN DE LLUVIA MH-RD/YL-38

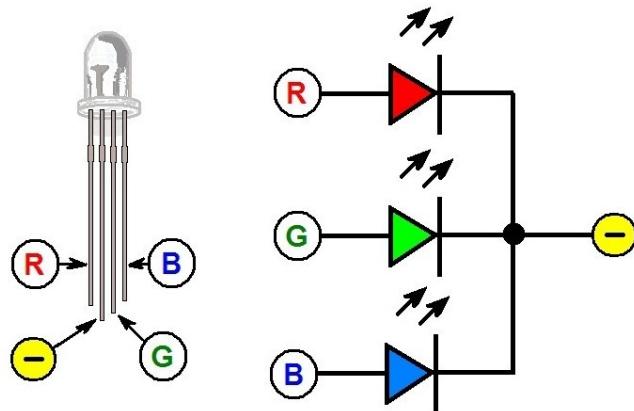


MODULO DE TIEMPO REAL TINY RTC DS1307

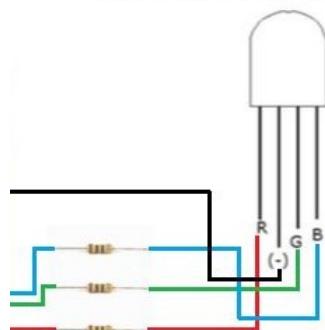


INDICADORES LUMINOSOS

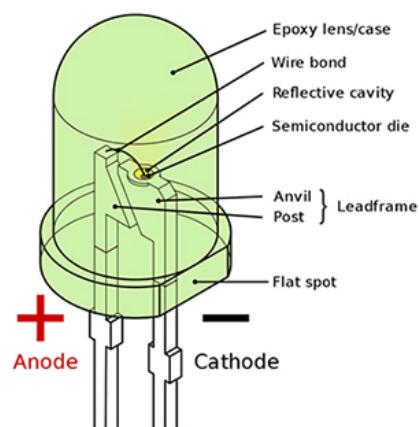
LED RGB (diodo emisor de luz)



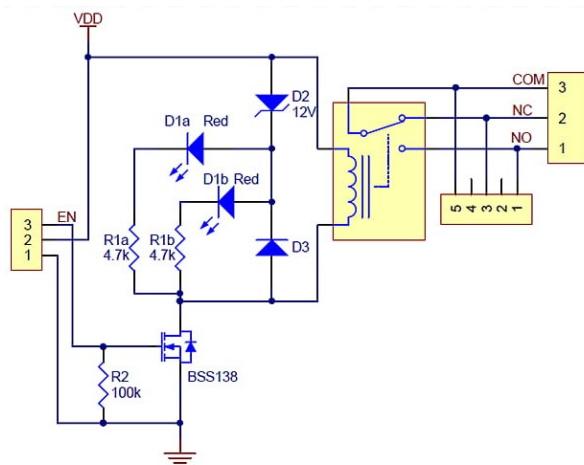
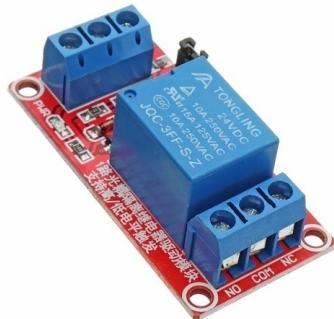
Common Cathode RGB LED



LED NORMAL



PRE-ACTUADOR RELE DE UN CANAL TONGLING JQC-3FF-S-Z



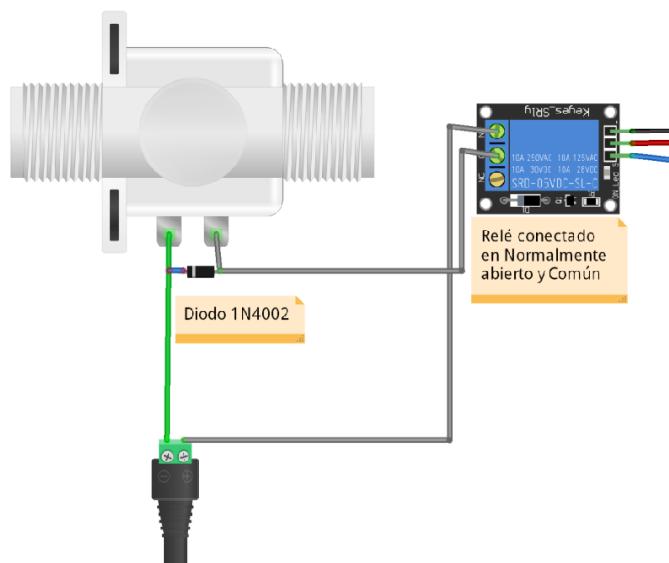
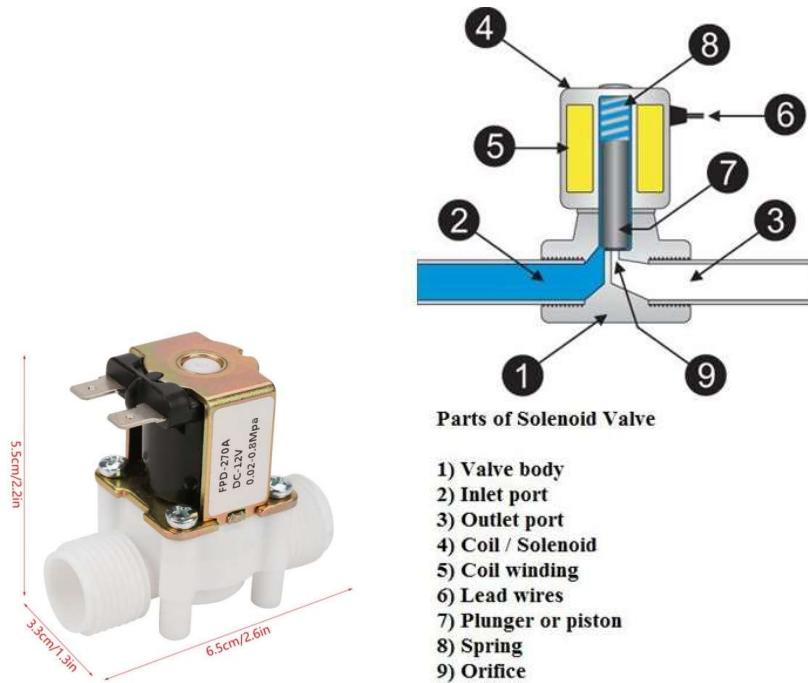
Entradas:

- DC+: positive power supply (VCC)
- DC-: negative power supply (GND)
- IN: control del relé, puede ser high o low.

Salidas:

- NO: normalmente abierto.
- COM: Común
- NC: normalmente cerrado.

ACTUADOR ELECTROVALVULA FPD-270A



Anexo D

Aplicativos de video del Sistema Operativo Linux OpenWrt Fswebcam save image from video stream

Details:

Written by Linux OpenWrt

Install Software:

```
opkg update
opkg install fswebcam
```

```
fswebcam --help
Usage: fswebcam [<options>] <filename> [<options>] <filename> ... ]
```

Options:

-?, --help	Display this help page and exit.
-c, --config <filename>	Load configuration from file.
-q, --quiet	Hides all messages except for errors.
-v, --verbose	Displays extra messages while capturing
--version	Displays the version and exits.
-l, --loop <seconds>	Run in loop mode.
-b, --background	Run in the background.
-o, --output <filename>	Output the log to a file.
-d, --device <name>	Sets the source to use.
-i, --input <number/name>	Selects the input to use.
-t, --tuner <number>	Selects the tuner to use.
-f, --frequency <number>	Selects the frequency use.
-p, --palette <name>	Selects the palette format to use.
-D, --delay <number>	Sets the pre-capture delay time. (seconds)
-r, --resolution <size>	Sets the capture resolution.
--fps <framerate>	Sets the capture frame rate.
-F, --frames <number>	Sets the number of frames to capture.
-S, --skip <number>	Sets the number of frames to skip.
--dumpframe <filename>	Dump a raw frame frame to file.
-s, --set <name><value>	Sets a control value.
--revert	Restores original captured image.
--flip <direction>	Flips the image. (h, v)
--crop <size>[,<offset>]	Crop a part of the image.
--scale <size>	Scales the image.
--rotate <angle>	Rotates the image in right angles.
--deinterlace	Reduces interlace artifacts.
--invert	Inverts the images colours.
--greyscale	Removes colour from the image.
--swapchannels <c1c2>	Swap channels c1 and c2.
--no-banner	Hides the banner.
--top-banner	Puts the banner at the top.
--bottom-banner	Puts the banner at the bottom. (Default)
--banner-colour <colour>	Sets the banner colour. (#AARRGGBB)
--line-colour <colour>	Sets the banner line colour.
--text-colour <colour>	Sets the text colour.
--font <[name][:size]>	Sets the font and/or size.
--no-shadow	Disables the text shadow.
--shadow	Enables the text shadow.
--title <text>	Sets the main title. (top left)
--no-title	Clears the main title.
--subtitle <text>	Sets the sub-title. (bottom left)
--no-subtitle	Clears the sub-title.
--timestamp <format>	Sets the timestamp format. (top right)
--no-timestamp	Clears the timestamp.
--gmt	Use GMT instead of local timezone.
--info <text>	Sets the info text. (bottom right)
--no-info	Clears the info text.
--underlay <PNG image>	Sets the underlay image.
--no-underlay	Clears the underlay.
--overlay <PNG image>	Sets the overlay image.
--no-overlay	Clears the overlay.
--jpeg <factor>	Outputs a JPEG image. (-1, 0 - 95)
--png <factor>	Outputs a PNG image. (-1, 0 - 10)
--save <filename>	Save image to file.
--exec <command>	Execute a command and wait for it to complete.

Save image from video stream:

```
fswebcam -i 0 -d v4l2:/dev/video0 --jpeg 95 --save test.jpg -S 20 -r 640x480
```

Save image at test.jpg, skip the first (group) frames with the -S 20.

Replace "test.jpg" with `date +"%G%m%d%H%M%S".jpg

Mjpeg stream video

Details:

Written by Linux OpenWrt

Install Software:

```
opkg update  
opkg install mjpg-streamer
```

mjpg-streamer output plugin:

```
output_http
```

This plugin streams JPEG data from input plugins via HTTP.

Usage:

```
mjpg_streamer [input plugin options] -o 'output_http.so [options]'
```

The following parameters can be passed to this plugin:

```
[ -w | --www ]: folder that contains webpages in flat hierarchy (no subfolders)  
[ -p | --port ]: TCP port for this HTTP server  
[ -c | --credentials ]: ask for "username:password" on connect  
[ -n | --nocmds ]: disable execution of commands
```

Browser/VLC

To view the stream use VLC or Firefox/Chrome and open the URL:

```
http://127.0.0.1:8080/?action=stream
```

If there are multiple input plugins, you can access each stream individually:

```
http://127.0.0.1:8080/?action=stream_0
```

```
http://127.0.0.1:8080/?action=stream_1
```

To do the same as the GET request above using NSURLConnection in Objective-C, a POST request seems to work:

```
POST http://127.0.0.1:8080/stream
```

To view a single JPEG just open this URL:

```
http://127.0.0.1:8080/?action=snapshot
```

mplayer

To play the HTTP M-JPEG stream with mplayer:

```
# mplayer -fps 30 -demuxer lavf http://127.0.0.1:8080/?action=stream&ignored.mjpg
```

It might be necessary to configure mplayer to prefer IPv4 instead of IPv6:

```
# vi ~/mplayer/config
```

add or change the option:

```
prefer-ipv4=yes
```