



REPÚBLICA DEL ECUADOR

**Escuela Politécnica Nacional**

" E S C I E N T I A H O M I N I S S A L U S "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

***Respeto hacia sí mismo y hacia los demás.***

# **ESCUELA POLITÉCNICA NACIONAL**

**FACULTAD DE INGENIERÍA ELÉCTRICA Y  
ELECTRÓNICA**

**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO INALÁMBRICO  
UTILIZANDO EL ESTANDAR IEEE 802.15.4 “ZIGBEE” PARA EL  
MONITOREO Y ADQUISICIÓN DE DATOS DE LOS SENSORES  
METEOROLÓGICOS DEL INAMHI**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
ELECTRÓNICA Y TELECOMUNICACIONES**

**MIGUEL ANGEL BONILLA VERGARA**  
**bonillamg@gmail.com**

**DIRECTOR: ING. TARQUINO FABIÁN SÁNCHEZ ALMEIDA**  
**tarquino.sanchez@epn.edu.ec**

**Quito, marzo 2014**

## DECLARACIÓN

Yo Miguel Angel Bonilla Vergara, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

Miguel Angel Bonilla Vergara

## CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Miguel Angel Bonilla Vergara, bajo mi supervisión.

---

**Ing. Tarquino Sánchez**  
**DIRECTOR DEL PROYECTO**

## **AGRADECIMIENTOS**

Agradezco a mis queridos padres Marquito y Anita quienes siempre me apoyan incondicionalmente. Agradezco a mis ñaños y a todos los miembros de mi familia Bonilla, Vergara, Pazmiño y Melo que al son de ¿Qué fue esa Tesis? han sido un apoyo permanente y que sin los cuales este proyecto no sería posible. De igual manera, a toda la manada de amigos de la universidad autocalificados como los hommies, quienes siempre han estado pendientes, y con quienes he compartido incontables desventuras y aventuras. A mis amigos del alemán y de la Católica que siempre me han dado ánimos. A mi tutor por ser comprensivo y ayudarme con la culminación de este proyecto. Y por supuesto la ayuda del INAMHI por soportarme todo este largo tiempo, y muy especial a los amigos del IDT que siempre me ayudaron y apoyaron.

Muchas gracias a todos ustedes!

## **DEDICATORIA**

Dedico el presenta trabajo a mis queridos padres Anita y Marquito.

## CONTENIDO

ÍNDICE DE FIGURAS.....	ix
ÍNDICE DE TABLAS.....	xviii
RESUMEN.....	xxi
PRESENTACIÓN.....	xxii
<b>CAPÍTULO 1. MARCO TEÓRICO.....</b>	<b>1</b>
<b>1.1. DESCRIPCIÓN DE LOS ESTANDARES DE COMUNICACIONES</b>	
<b>IEEE 802.15.4 Y ZIGBEE.....</b>	<b>1</b>
1.1.1. INTRODUCCIÓN AL ESTANDAR ZIGBEE.....	1
1.1.2. CONCEPTOS DE COMUNICACIÓN ZIGBEE E IEEE 802.15.4.....	5
<b>1.2. ARQUITECTURA ZIGBEE.....</b>	<b>9</b>
1.2.1. CAPAS DE LA ARQUITECTURA ZIGBEE E IEEE802.15.4.....	9
1.2.2. CAPA FÍSICA IEEE 802.15.4 ( <i>PHYSICAL LAYER PHY</i> ).....	10
1.2.3. CAPA DE ACCESO AL MEDIO IEEE 802.15.4 ( <i>MEDIUM ACCESS</i> <i>CONTROL MAC</i> ).....	13
1.2.4. CAPA DE RED ZIGBEE NWK ( <i>NETWORK LAYER</i> ).....	18
1.2.5. CAPA DE APLICACIÓN APL ( <i>APPLICATION LAYER</i> ).....	23
<b>1.3. DESCIPCIÓN GENERAL DE LAS TECNOLGÍAS GSM Y GPRS.....</b>	<b>23</b>
1.3.1. DESCRIPCIÓN GENERAL DEL SISTEMA GLOBAL DE COMUNICACIÓN MOVIL ( <i>GLOBAL SYSTEM FOR MOBILE</i> <i>COMMUNICATION GSM</i> ).....	23
1.3.2. OBJETIVOS, VENTAJAS Y CONCEPTOS GENERALES DEL SERVICIO DE PAQUETES DE RADIO ( <i>GENERAL PACKET RADIO</i> <i>SERVICE GPRS</i> ).....	25
1.3.3. ARQUITECTURA GPRS.....	27
1.3.4. PROCEDIMIENTO DE CONEXIÓN GPRS.....	36
1.3.5. PROTOCOLO DE CONTROL DE TRANSMISIÓN ( <i>TRANSMISSION</i> <i>CONTROL PROTOCOL TCP</i> ) EN GPRS.....	37

<b>CAPÍTULO 2. REQUERIMIENTOS DEL PROTOTIPO.....</b>	<b>41</b>
<b>2.1. DESCRIPCIÓN GENERAL DEL INAMHI.....</b>	<b>41</b>
2.1.1. FUNCIÓN DEL INAMHI.....	41
2.1.2. ORGANIGRAMA DEL INAMHI.....	42
2.1.3. CLASIFICACIÓN DE LAS ESTACIONES HIDROMETEOROLÓGICAS UTILIZADAS POR EL INAMHI.....	43
2.1.4. RED DE ESTACIONES AUTOMÁTICAS METEOROLÓGICAS INSTALADAS.....	46
<b>2.2. NECESIDAD Y SOLUCIÓN REFERENTE A LA IMPLEMENTACIÓN DE UNA ESTACIÓN AUTOMÁTICA INALÁMBRICA POR PARTE DEL INAMHI.....</b>	<b>47</b>
<b>2.3. DETERMINACIÓN DE LAS ESPECIFICACIONES DEL PROTOTIPO....</b>	<b>48</b>
2.3.1. TOPOLOGÍA DE RED DEL PROTOTIPO.....	48
2.3.2. FUNCIONES Y DEFINICIÓN DE LOS MÓDULOS DEL PROTOTIPO....	49
2.3.3. VARIABLES DE ENTRADA Y SALIDA DE CADA UNO DE LOS MÓDULOS DEL PROTOTIPO.....	51
2.3.4. ESPECIFICACIONES DEL PROTOTIPO.....	55
<b>2.4. REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA IMPLEMENTACIÓN DEL PROTOTIPO.....</b>	<b>56</b>
2.4.1. DISPOSITIVOS Y CARACTERÍSTICAS TÉCNICAS DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.....	56
2.4.2. DISPOSITIVOS Y CARACTERÍSTICAS TÉCNICAS DEL MÓDULO DE PRECIPITACIÓN.....	59
2.4.3. DISPOSITIVOS Y CARACTERÍSTICAS TÉCNICAS DEL MÓDULO DE ADQUISICIÓN DE DATOS.....	60
2.4.4. DISPOSITIVOS Y CARACTERÍSTICAS TÉCNICAS DEL MÓDULO DE TRANSMISIÓN Y PROCESAMIENTOS DE DATOS GPRS.....	60
2.4.5. REQUERIMIENTOS DE SOFTWARE PARA IMPLEMENTACIÓN DEL PROTOTIPO.....	61
<b>CAPÍTULO 3. DISEÑO DEL PROTOTIPO.....</b>	<b>63</b>

<b>3.1. PROCEDIMIENTO DE DISEÑO DEL PROTOTIPO.....</b>	<b>63</b>
<b>3.2. DISEÑO DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.....</b>	<b>64</b>
3.2.1. SISTEMAS Y ELEMENTOS DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.....	64
3.2.2. DESCRIPCIÓN Y DISEÑO DEL SISTEMA DE PROCESAMIENTO Y PRESENTACIÓN DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.....	64
3.2.3. DESCRIPCIÓN Y DISEÑO DEL SISTEMA DE ENERGIZACIÓN DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.....	101
3.2.4. DESCRIPCIÓN Y DISEÑO DEL SISTEMA DE REGULACIÓN Y DETECCIÓN DE VOLTAJE DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.....	105
3.2.5. ESQUEMA TOTAL DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.....	116
<b>3.3. DISEÑO DEL MÓDULO DE PRECIPITACIÓN.....</b>	<b>118</b>
3.3.1. SISTEMAS Y ELEMENTOS DEL MÓDULO DE PRECIPITACIÓN.....	118
3.3.2. DESCRIPCIÓN Y DISEÑO DEL SISTEMA DE PROCESAMIENTO DEL MÓDULO DE PRECIPITACIÓN.....	119
3.3.3. ESQUEMA TOTAL DEL MÓDULO DE PRECIPITACIÓN.....	127
<b>3.4. DISEÑO DEL MÓDULO DE ADQUISICIÓN DE DATOS.....</b>	<b>129</b>
3.4.1. SISTEMAS Y ELEMENTOS DEL MÓDULO DE ADQUISICIÓN DE DATOS.....	130
3.4.2. DESCRIPCIÓN DEL DISPOSITIVO DE TX/RX ZIGBEE FFD.....	131
3.4.3. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE RECEPCIÓN Y TRANSMISIÓN SERIAL API.....	133
3.4.4. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE ADQUISICIÓN DE DATOS DE TEMPERATURA Y HUMEDAD RELATIVA Y NIVEL DE VOLTAJE DE BATERÍA A.....	144
3.4.5. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE ADQUISICIÓN DE DATOS DE PRECIPITACIÓN Y NIVEL DE VOLTAJE DE BATERÍA B.....	150

3.4.6. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE PRESENTACIÓN DE DATOS METEOROLÓGICOS.....	151
3.4.7. VENTANA DE INFORMACIÓN METEOROLÓGICA.....	160
<b>3.5. DISEÑO DEL MÓDULO DE TRANSMISIÓN Y PROCESAMIENTO DE DATOS GPRS.....</b>	<b>164</b>
3.5.1. SISTEMAS Y ELEMENTOS DEL MÓDULO DE TRANSMISIÓN Y PROCESAMIENTO DE DATOS GPRS.....	166
3.5.2. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE TRANSMISIÓN DE DATOS GPRS.....	166
3.5.3. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE TRANSMISIÓN Y RECEPCIÓN WAVECOM FASTRACK.....	168
3.5.4. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE RECEPCIÓN DE DATOS GPRS.....	172
3.5.5. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE RECOPIACIÓN DE DATOS GPRS.....	173
3.5.6. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE PRESENTACIÓN DE DATOS GPRS.....	173
3.5.7. VENTANA DE USUARIO.....	175
<b>CAPÍTULO 4. IMPLEMENTACIÓN Y PRUEBAS DEL PROTOTIPO.....</b>	<b>179</b>
<b>4.1. DISEÑO DE PLACAS.....</b>	<b>179</b>
4.1.1. DIAGRAMAS ESQUEMÁTICOS DE LAS PLACAS DISEÑADAS.....	179
<b>4.2. CALIBRACIÓN DE LOS SENSORES.....</b>	<b>179</b>
4.2.1. CALIBRACIÓN DEL SENSOR DE TEMPERATURA HMP45A.....	182
4.2.2. CALIBRACIÓN DEL SENSOR DE HUMEDAD HMP45A.....	183
4.2.3. CALIBRACIÓN DEL SENSOR DE PRECIPITACIÓN TR-525M.....	185
<b>4.3. PRUEBAS Y RESULTADOS.....</b>	<b>187</b>
4.3.1. PRUEBAS DE RECEPCIÓN RSSI.....	187
4.3.2. PRUEBAS DE FUNCIONAMIENTO DEL PROTOTIPO.....	189
4.3.3. PRUEBAS DE VALIDACIÓN DE INFORMACIÓN DE TEMPERATURA, HUMEDAD Y PRECIPITACIÓN.....	191
<b>4.4. INSTALACIÓN DEL PROTOTIPO EN CAMPO.....</b>	<b>194</b>

<b>4.5. COSTOS REFERENCIALES DEL PROTOTIPO.....</b>	<b>196</b>
4.5.1. COSTO REFERENCIAL DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.....	198
4.5.2. COSTO REFERENCIAL DEL MÓDULO DE PRECIPITACIÓN.....	198
4.5.3. COSTO REFERENCIAL DEL MÓDULO DE ADQUISICIÓN DE DATOS.....	198
4.5.4. COSTO REFERENCIAL DEL MÓDULO DE TRANSMISIÓN Y PROCESAMIENTO DE DATOS GPRS Y DE SOFTWARE DEL PROTOTIPO.....	198
4.5.5. COSTO TOTALES REFERENCIAL DEL PROTOTIPO.....	199

## **CAPÍTULO 5. CONCLUSIONES Y RECOMENDACIONES..... 200**

<b>5.1. CONCLUSIONES.....</b>	<b>200</b>
5.1.1. CONCLUSIONES DE IMPLEMENTACIÓN, DISEÑO, VENTAJAS Y DESVENTAJAS DEL PROTOTIPO.....	200
5.1.2. CONCLUSIONES DE PRUEBAS Y RESULTADOS DEL PROTOTIPO.....	201
<b>5.2. RECOMENDACIONES.....</b>	<b>203</b>

## **REFERENCIAS BIBLIOGRÁFICAS..... 204**

### **ANEXOS**

- A. PROGRAMAS DE PROCESAMIENTO DE INFORMACIÓN DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA.**
- B. PROGRAMAS DE PROCESAMIENTO DE INFORMACIÓN DEL MÓDULO DE PRECIPITACIÓN.**
- C. DIAGRAMAS DE BLOQUES DE LOS PROGRAMAS PARA LA ADQUISICIÓN, TRANSMISIÓN Y RECEPCIÓN DE VARIABLES METEOROLÓGICAS DE TEMPERATURA, HUMEDAD Y PRECIPITACIÓN.**

## ÍNDICE DE FIGURAS

Figura 1.1	<i>Tipos y Roles de los Dispositivos ZigBee</i> .....	3
Figura 1.2	<i>Topología tipo Estrella</i> .....	4
Figura 1.3	<i>Topología tipo Malla</i> .....	4
Figura 1.4	<i>Transferencia de Datos desde un Coordinador a un Dispositivo Final: (a) Red Beacon y (b) Red No-Beacon</i> .....	7
Figura 1.5	<i>Transferencia de Datos a un Coordinador en el Esquema IEEE 802.15.4: (a) Red Beacon y (b) Red No-Beacon</i> .....	8
Figura 1.6	<i>Detalle de las Capas y Arquitectura de una Red ZigBee</i> .....	10
Figura 1.7	<i>Formato del Protocolo de la Unidad de Datos de Capa Física (PHY Protocol Data Unit PPDU)</i> .....	12
Figura 1.8	<i>Formatos de una Super-Trama</i> .....	14
Figura 1.9	<i>(a) Formato General de Trama MAC y (b) Detalles del Campo de Control</i> .....	15
Figura 1.10	<i>Ejemplo de una División Simple de Módulo 2</i> .....	17
Figura 1.11	<i>Formato Trama de Datos de la Capa MAC</i> .....	17
Figura 1.12	<i>Formato de Trama de Acuse de Recibo (ACK)</i> .....	18
Figura 1.13	<i>Redes de: (a) Difusión, (b) Punta a Multipunto y (c) Punto a punto</i> .....	19
Figura 1.14	<i>Enrutamiento Punto a Multipunto iniciado por un Dispositivo que es Miembro de un Grupo de Difusión Múltiple</i> .....	20
Figura 1.15	<i>Relación entre Dispositivos Padres e Hijos en una Topología de Árbol</i> .....	20
Figura 1.16	<i>Análisis del Costo del Enlace</i> .....	22
Figura 1.17	<i>Aplicaciones y Alcances GPRS</i> .....	26
Figura 1.18	<i>Servicio de Transmisión GPRS</i> .....	27
Figura 1.19	<i>Separación Física de la Sección de Acceso de la Red del Núcleo GSM y GPRS</i> .....	28
Figura 1.20	<i>Subsistema de Acceso</i> .....	28
Figura 1.21	<i>Estación Móvil GSM/GPRS</i> .....	29
Figura 1.22	<i>Esquema del Subsistema de Acceso GPRS</i> .....	31
Figura 1.23	<i>Componentes del Núcleo de una Red GPRS</i> .....	32

<i>Figura 1.24</i>	<i>Inicio de Sesión de un MS en un GGSN.....</i>	<i>34</i>
<i>Figura 1.25</i>	<i>Tres Posibilidades de Interconexión entre la PCU y la SGSN.....</i>	<i>35</i>
<i>Figura 1.26</i>	<i>Estructura Típica de una Red Gn.....</i>	<i>36</i>
<i>Figura 1.27</i>	<i>Procedimientos GPRS.....</i>	<i>37</i>
<i>Figura 1.28</i>	<i>Conjunto de Protocolos compatibles con IP.....</i>	<i>38</i>
<i>Figura 1.29</i>	<i>Cabecera TCP.....</i>	<i>39</i>
<i>Figura 2.1</i>	<i>Organigrama del Instituto Nacional de Meteorología e Hidrología del Ecuador INAMHI.....</i>	<i>42</i>
<i>Figura 2.2</i>	<i>Diagrama de Clasificación de las Estaciones Hidrometeorológicas.....</i>	<i>43</i>
<i>Figura 2.3</i>	<i>Estación Climatológica Convencional.....</i>	<i>44</i>
<i>Figura 2.4</i>	<i>Caseta Meteorológica.....</i>	<i>45</i>
<i>Figura 2.5</i>	<i>Partes de una Estación Meteorológica Campbell.....</i>	<i>46</i>
<i>Figura 2.6</i>	<i>Topología tipo Estrella de la Red Inalámbrica del Prototipo.....</i>	<i>49</i>
<i>Figura 2.7</i>	<i>Variables de Entrada y Salida del Prototipo.....</i>	<i>52</i>
<i>Figura 2.8</i>	<i>Funciones y Variables de Entrada y Salida del Módulo de Temperatura y Humedad Relativa.....</i>	<i>52</i>
<i>Figura 2.9</i>	<i>Funciones y Variables de Entrada y Salida del Módulo de Precipitación.....</i>	<i>53</i>
<i>Figura 2.10</i>	<i>Funciones y Variables de Entrada y Salida del Módulo de Adquisición de Datos.....</i>	<i>54</i>
<i>Figura 2.11</i>	<i>Funciones y Variables de Entrada y Salida del Módulo de Transmisión y Procesamiento de Datos GPRS.....</i>	<i>55</i>
<i>Figura 2.12</i>	<i>Sensor de Temperatura y Humedad Relativa HMP45A/D de VAISALA.....</i>	<i>57</i>
<i>Figura 2.13</i>	<i>Microcontrolador ATmega164P del Fabricante Atmel.....</i>	<i>58</i>
<i>Figura 2.14</i>	<i>Módulo de Tx/Rx XBEE-PRO.....</i>	<i>59</i>
<i>Figura 2.15</i>	<i>Sensores de Precipitación de la Serie 525.....</i>	<i>59</i>
<i>Figura 2.16</i>	<i>Modem de Transmisión GPRS Fastrack Wavecom.....</i>	<i>61</i>
<i>Figura 3.1</i>	<i>Esquema del Prototipo Inalámbrico utilizando el Estándar ZigBee y Transmisión GPRS para el Monitoreo y Adquisición de Datos Meteorológicos.....</i>	<i>63</i>

Figura 3.2	Diagrama de Bloques del Módulo de Temperatura y Humedad Relativa.....	65
Figura 3.3	Esquemas que muestran: (a) Conjunto de Programas del Módulo de Temperatura y Humedad Relativa y (b) Conjunto de Parámetros a ser Configurados en el Módulo XBee-PRO RFD.....	66
Figura 3.4	Diagrama de Bloques de la Conversión ADC.....	67
Figura 3.5	Diagrama de Bloques de una Señal Acondicionada para la correspondiente Conversión ADC.....	67
Figura 3.6	Circuito Interfaz del Transductor.....	68
Figura 3.7	Valores de Resistencias del Circuito Interfaz del Transductor.....	71
Figura 3.8	Librería ADC.h dentro del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa.....	72
Figura 3.9	Diagrama de Flujo de la Función de Interrupción ADC ISR(ADC_vect).....	74
Figura 3.10	Conexión del Pin AVCC del ADC para Reducción del Ruido.....	74
Figura 3.11	Diagrama de Bloques de un Temporizador/Contador.....	75
Figura 3.12	Librería muestreo.h dentro del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa.....	76
Figura 3.13	Diagrama de Flujo de la Interrupción del Registro: (a) OCR1A y (b) OCR1B.....	78
Figura 3.14	Formato de Tramas UART.....	79
Figura 3.15	Librería UART.h dentro del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa.....	79
Figura 3.16	Diagramas de Flujo de las Sigüientes Funciones: (a) Envío de Datos del Sensor de Temperatura, (b) Envío de Datos del Sensor de Humedad y (c) Envío del Porcentaje de Nivel de Voltaje A para Funcionamiento del Circuito.....	82
Figura 3.17	Diagrama de Flujo de la Función de Interrupción de Recepción Activada cuando se Detecta que el Buffer Rx Contiene un Carácter de Datos.....	82
Figura 3.18	Librería LCD.h del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa.....	83

<i>Figura 3.19</i>	<i>Divisor Resistivo para Manejo de Voltaje para Control del Contraste del LCD.....</i>	<i>84</i>
<i>Figura 3.20</i>	<i>Secuencia de Eventos del Proceso de una Interrupción Externa....</i>	<i>86</i>
<i>Figura 3.21</i>	<i>Programa de Interrupciones Externas dentro del Esquema del Conjunto de Programas del Módulo de Precipitación.....</i>	<i>86</i>
<i>Figura 3.22</i>	<i>Función de Interrupción Externa INT1 para Encendido de la Pantalla LCD.....</i>	<i>87</i>
<i>Figura 3.23</i>	<i>Programa de Procesamiento de Información dentro del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa.....</i>	<i>88</i>
<i>Figura 3.24</i>	<i>Diagrama de Flujo de la Función Principal del Programa del Módulo de Temperatura y Humedad Relativa.....</i>	<i>91</i>
<i>Figura 3.25</i>	<i>Función de Lectura del Voltaje: (a) Sensor de Temperatura y (b) Sensor de Humedad.....</i>	<i>92</i>
<i>Figura 3.26</i>	<i>Función de la Lectura del Nivel de Voltaje Necesario para Funcionamiento del Circuito.....</i>	<i>93</i>
<i>Figura 3.27</i>	<i>Ventanas del Software X-CTU que muestran los Parámetros de Configuración del Módulo RFD ZigBee de Temperatura y Humedad Relativa.....</i>	<i>101</i>
<i>Figura 3.28</i>	<i>Panel Solar de 12 V con un ISC de 1.6 A.....</i>	<i>102</i>
<i>Figura 3.29</i>	<i>Batería CSB de 12 V 7 Ah.....</i>	<i>103</i>
<i>Figura 3.30</i>	<i>Controlador de Carga Solar Sun Saver-6.....</i>	<i>103</i>
<i>Figura 3.31</i>	<i>Diagrama de Bloques del Circuito de Regulación de 5 V que Alimenta un Microcontrolador Atmega164P.....</i>	<i>106</i>
<i>Figura 3.32</i>	<i>Diagrama Circuito Integrado LM7805 para Regulación de Voltaje de 5 V.....</i>	<i>106</i>
<i>Figura 3.33</i>	<i>XBee-Regulator de SparkFun.....</i>	<i>107</i>
<i>Figura 3.34</i>	<i>Esquemas de: (a) Circuito de Conexiones del Dispositivo XBEE, (b) Circuito Integrado Regulador de 3.5 V y (c) Circuitos Leds de Indicación.....</i>	<i>107</i>
<i>Figura 3.35</i>	<i>Diagrama de Bloques del Circuito de Detección de Voltaje.....</i>	<i>108</i>
<i>Figura 3.36</i>	<i>Esquemas de: (a) Circuito de Protección de Corriente "Foldback" y (b) Curva de Limitación de Corriente.....</i>	<i>109</i>

<i>Figura 3.37</i>	<i>Circuito Divisor de Tensión</i> .....	112
<i>Figura 3.38</i>	<i>Circuitos de Protección y Detección de Voltaje</i> .....	114
<i>Figura 3.39</i>	<i>Diagrama Esquemático del Módulo de Temperatura y Humedad Relativa</i> .....	117
<i>Figura 3.40</i>	<i>Diagrama de Bloques del Módulo de Precipitación</i> .....	118
<i>Figura 3.41</i>	<i>Esquemas que muestran: (a) Conjunto de Programas del Módulo de Precipitación y (b) Conjunto de Parámetros a ser Configurados en el Dispositivo XBee-PRO RFD</i> .....	120
<i>Figura 3.42</i>	<i>Diagramas de Flujo de: (a) Función de Interrupción Externa INT0 para Conteo de Basculaciones del Sensor de Precipitación y (b) Función de Lectura de Pulsos del Reed Switch del Sensor de Precipitación para Obtener Valores de Precipitación en (mm)</i> .....	121
<i>Figura 3.43</i>	<i>Programa de Procesamiento de Información dentro del Esquema del Conjunto de Programas del Módulo de Precipitación</i> .....	122
<i>Figura 3.44</i>	<i>Función Principal del Procesamiento de Señales del Sensor de Precipitación</i> .....	124
<i>Figura 3.45</i>	<i>Función de la Lectura del Nivel de Voltaje Necesario para Funcionamiento del Circuito</i> .....	125
<i>Figura 3.46</i>	<i>Diagrama Esquemático del Módulo de Precipitación</i> .....	128
<i>Figura 3.47</i>	<i>Diagrama de Bloques del Proceso de Diseño del Módulo de Adquisición de Datos</i> .....	129
<i>Figura 3.48</i>	<i>Sistemas y Elementos del Módulo de Adquisición de Datos</i> .....	130
<i>Figura 3.49</i>	<i>Comunicación entre el Adquisidor de Datos y el Módulo de Tx/Rx ZigBee FFD</i> .....	131
<i>Figura 3.50</i>	<i>Ventanas del Software X-CTU que Muestran los Parámetros Configurados del Dispositivo Coordinador FFD ZigBee del Módulo de Adquisición de Datos</i> .....	133
<i>Figura 3.51</i>	<i>Formato de la Trama API sin Caracteres de Espacio</i> .....	134
<i>Figura 3.52</i>	<i>Sistemas del Módulo de Transmisión y Recepción de Tramas API</i> .....	135
<i>Figura 3.53</i>	<i>Diagrama de Bloques del Interfaz VISA de LabView</i> .....	136

<i>Figura 3.54</i>	<i>Trama de un Paquete de Recepción de 64-bits de Direccionamiento.....</i>	<i>137</i>
<i>Figura 3.55</i>	<i>Diagrama de Flujo de la Función de Filtrado de Tramas API Rx de 64-bits de Direccionamiento.....</i>	<i>138</i>
<i>Figura 3.56</i>	<i>Diagrama de Flujo de la Función de Chequeo del Campo de Checksum de la Trama API Rx de 64-bits.....</i>	<i>139</i>
<i>Figura 3.57</i>	<i>Diagrama de Flujo de Funciones de Verificación del Campo de Datos RF y de Extracción de los Strings de Datos tanto del Módulo de Temperatura y Humedad Relativa como del Módulo de Precipitación.....</i>	<i>140</i>
<i>Figura 3.58</i>	<i>Trama API de Transmisión de 64-bits de Direccionamiento.....</i>	<i>141</i>
<i>Figura 3.59</i>	<i>Estructura de una Trama API Tx de 64-bits de Direccionamiento que Incluye Datos de Temperatura.....</i>	<i>144</i>
<i>Figura 3.60</i>	<i>Diagrama de Bloques del Módulo de Adquisición de Datos de Temperatura, Humedad Relativa y Nivel de Voltaje de Batería A.....</i>	<i>145</i>
<i>Figura 3.61</i>	<i>Diagramas de Flujo de las Funciones de Procesamiento y Muestreo de Información de Temperatura y Humedad Relativa....</i>	<i>146</i>
<i>Figura 3.62</i>	<i>Diagrama de Flujo de la Función de Procesamiento de Información del Nivel de Voltaje de Batería A.....</i>	<i>147</i>
<i>Figura 3.63</i>	<i>Proceso de Almacenamiento de la Información Muestreada en un Archivo de Texto.....</i>	<i>148</i>
<i>Figura 3.64</i>	<i>Diagrama de Bloques del Módulo de Aquisición de Datos de Precipitación y Nivel de Voltaje de Batería B.....</i>	<i>149</i>
<i>Figura 3.65</i>	<i>Diagramas de Flujo de la Función de Procesamiento y Muestreo de Información de Precipitación.....</i>	<i>150</i>
<i>Figura 3.66</i>	<i>Diagrama de Bloques del Módulo de Presentación de Datos Meteorológicos.....</i>	<i>152</i>
<i>Figura 3.67</i>	<i>Diagrama de Bloques del Subprograma de Creación de un Archivo de Texto que contiene Información de Temperatura, Humedad y Precipitación, ordenada y muestreada por Fecha y Hora en Orden Descendente.....</i>	<i>154</i>

<i>Figura 3.68 Diagrama de Flujo para Obtención de un String de Información que Permita Extraer los Datos Muestreados de Temperatura, Humedad o Precipitación de los Archivos de Texto en Base a la Información Ingresada por el Usuario .....</i>	<i>155</i>
<i>Figura 3.69 Diagrama de Flujo para Organización de un Arreglo de Directorios que Indiquen la Dirección de los Archivos de Texto de los cuales se Extrae los Datos Muestreados en Arreglos de Datos.....</i>	<i>156</i>
<i>Figura 3.70 Arreglos de Datos Organizado por Fecha y Hora de: (a) Temperatura y Humedad (b) Precipitación.....</i>	<i>157</i>
<i>Figura 3.71 Diagrama de Flujo para Presentar la Información Individual de una sola Fecha Ingresada por el Usuario.....</i>	<i>158</i>
<i>Figura 3.72 Presentación de la Información Individual de una sola Fecha Ingresada por el Usuario de: (a) Temperatura y Humedad Relativa y (b) Precipitación.....</i>	<i>158</i>
<i>Figura 3.73 Ventana de Inicio para Presentación de Información General del Módulo de Adquisición de Datos.....</i>	<i>160</i>
<i>Figura 3.74 Pestañas del Módulo de Adquisición de Datos de Temperatura y Humedad: (a) Información de Temperatura y Humedad Relativa, (b) Gráficas de Temperatura y Humedad Relativa y (c) Tabla de Valores Adquiridos.....</i>	<i>161</i>
<i>Figura 3.75 Pestañas del Módulo de Adquisición de Datos de Precipitación: (a) Información y Gráfica de Precipitación, (b) Tabla de Valores Adquiridos.....</i>	<i>162</i>
<i>Figura 3.76 Pestañas del Módulo de Presentación del Datos Meteorológicos: (a) Selección de Información Meteorológica, (b) Gráficas de Información Meteorológica.....</i>	<i>163</i>
<i>Figura 3.77 Solución M2M para Conexión Remota de Dos Maquinas Punto a Punto Utilizando el Protocolo TCP-IP.....</i>	<i>164</i>
<i>Figura 3.78 Protocolo TCP Orientado a la Conexión y Datagramas IP no Orientado a la Conexión.....</i>	<i>165</i>
<i>Figura 3.79 Modelo Cliente-Servidor.....</i>	<i>165</i>

<i>Figura 3.80</i>	<i>Sistemas y Elementos del Módulo de Transmisión y Procesamiento de Datos GPRS.....</i>	<i>166</i>
<i>Figura 3.81</i>	<i>Diagrama de Estados para el Diseño del Módulo de Transmisión de Datos GPRS.....</i>	<i>167</i>
<i>Figura 3.82</i>	<i>Modelo Cliente-Servidor TCP/IP y Definición de Direcciones IP y Sockets.....</i>	<i>168</i>
<i>Figura 3.83</i>	<i>Secuencia de Eventos para Autenticación y Registro del Modem dentro de la Red GPRS.....</i>	<i>170</i>
<i>Figura 3.84</i>	<i>Diagrama de Estados para el Diseño del Módulo de Recepción de Datos GPRS.....</i>	<i>172</i>
<i>Figura 3.85</i>	<i>Diagrama de Flujo del Diseño del Módulo de Recopilación de Datos GPRS.....</i>	<i>174</i>
<i>Figura 3.86</i>	<i>Diagrama de Bloques del Módulo de Presentación de Datos GPRS.....</i>	<i>175</i>
<i>Figura 3.87</i>	<i>Proceso de Reubicación del Parámetro de Precipitación dentro del Arreglo de Temperatura y Humedad.....</i>	<i>176</i>
<i>Figura 3.88</i>	<i>Arreglo de Datos GPRS Final de Temperatura, Humedad y Precipitación Organizado por Fecha y Hora en Orden Descendente.....</i>	<i>176</i>
<i>Figura 3.89</i>	<i>Ventana de Inicio para Presentación de Información General del Módulo de Adquisición de Datos GPRS.....</i>	<i>177</i>
<i>Figura 3.90</i>	<i>Pestañas de la Ventana de Presentación de Datos GPRS donde se muestra: (a) Información de Temperatura, Humedad y Precipitación Ordenada por Fecha y Hora en Orden Descendente y (b) Gráficas de Temperatura, Humedad y Precipitación.....</i>	<i>178</i>
<i>Figura 4.1</i>	<i>(a) Diagrama Esquemático de la Cara Frontal del PCB del Módulo de Temperatura y Humedad Relativa y (b) Placa con los Correspondientes Elementos soldados del Módulo de Temperatura y Humedad Relativa.....</i>	<i>180</i>
<i>Figura 4.2</i>	<i>(a) Diagrama Esquemático de la Cara Frontal del PCB del Módulo de Precipitación y (b) Placa con los Correspondientes Elementos soldados del Módulo de Precipitación.....</i>	<i>181</i>

<i>Figura 4.3</i>	<i>Calibración del Sensor de Temperatura HMP45A, en el Laboratorio de Metrología del INAMHI, por medio de la utilización de una Cámara de Pozo Seco.....</i>	<i>182</i>
<i>Figura 4.4</i>	<i>Proceso de Calibración de Temperatura.....</i>	<i>183</i>
<i>Figura 4.5</i>	<i>Gráfica de Temperatura vs Voltaje y su correspondiente Ecuación Lineal de los Datos Obtenidos del Sensor de Temperatura HMP45A.....</i>	<i>184</i>
<i>Figura 4.6</i>	<i>Calibración del Sensor de Humedad HMP45A, en el Laboratorio de Metrología del INAMHI, por medio de la utilización de una Cámara de Humedad.....</i>	<i>184</i>
<i>Figura 4.7</i>	<i>Gráfica de Humedad vs Voltaje y su Correspondiente Ecuación Lineal de los Datos Obtenidos del Sensor HMP45A.....</i>	<i>185</i>
<i>Figura 4.8</i>	<i>Ventana de Verificación de Nivel de Señal RSSI del Software X-CTU.....</i>	<i>188</i>
<i>Figura 4.9</i>	<i>Instrumento Virtual de Transmisión y Recepción de Tramas API de Información de Temperatura, Humedad y Precipitación....</i>	<i>189</i>
<i>Figura 4.10</i>	<i>Transmisión de Información Meteorológica vía GPRS del Módulo de Adquisición de Datos.....</i>	<i>190</i>
<i>Figura 4.11</i>	<i>Recepción de Información Meteorológica vía GPRS en el Servidor localizado en el INAMHI Quito.....</i>	<i>190</i>
<i>Figura 4.12</i>	<i>Histograma de Comparación de Valores de Temperatura para obtención del Error Absoluto de la Muestra de Valores Observados.....</i>	<i>192</i>
<i>Figura 4.13</i>	<i>Histograma de Comparación de Valores de Humedad para obtención del Error Absoluto de la Muestra de Valores Observados.....</i>	<i>194</i>
<i>Figura 4.14</i>	<i>Caja de Protección del Módulo de Temperatura y Humedad Relativa como del Módulo de Precipitación.....</i>	<i>195</i>
<i>Figura 4.15</i>	<i>Módulos Instalados en campo de: (a) Temperatura y Humedad Relativa y (b) Precipitación.....</i>	<i>195</i>

## ÍNDICE DE TABLAS

Tabla 1.1	Velocidades de Transmisión del Estándar IEEE 802.15.4, Frecuencias de Operación, Modulación, Número de Canales y Métodos de Ensanchamiento del Espectro de Frecuencia.....	3
Tabla 1.2	Subcampo del Delimitador de Inicio (Start of Frame Delimitator SFD) con Excepción de las Tramas ACK de Capa Física.....	13
Tabla 1.3	Valores de Longitudes de Trama de acuerdo a los Diferentes Tipos de Campos de Carga útil de Capa Física.....	13
Tabla 2.1	Instrumentos de Lectura y Registradores de Variables Hidrometeorológicas.....	44
Tabla 2.2	Especificaciones de Equipos y Dispositivos necesarios, de acuerdo a las Funciones de cada Módulo del Prototipo.....	56
Tabla 2.3	Características Técnicas de los Sensores de Temperatura y Humedad Relativa.....	57
Tabla 2.4	Características Técnicas de Microcontroladores.....	58
Tabla 2.5	Características Técnicas de los Dispositivos Terminales ZigBee.....	58
Tabla 2.6	Características Técnicas del Sensor de Precipitación.....	59
Tabla 2.7	Características Técnicas para Ambientes Indoor y Outdoor del Adquisidor de Datos.....	60
Tabla 2.8	Características Técnicas de Equipos y Dispositivos del Módulo de Transmisión y Procesamiento de Datos GPRS.....	61
Tabla 3.1	Selección de la Fuente de Auto Activación ADC.....	73
Tabla 3.2	Descripción de los Modos de Generación de Formas de Onda Requeridos.....	76
Tabla 3.3	Ecuaciones para el Cálculo de la Velocidad de Transmisión y su Correspondiente Valor de Registro.....	80
Tabla 3.4	Tabla de Asignación de Pines de un LCD Hitachi.....	84
Tabla 3.5	Tabla de Funciones de los Bits de Lectura, Escritura y de Registros del LCD.....	84
Tabla 3.6	Control de la Generación de la Interrupción Externa en el Pin INTn.....	87

<i>Tabla 3.7</i>	<i>Opciones de Configuración de la Velocidad de Transmisión Serial mediante el Comando ATBD.....</i>	<i>94</i>
<i>Tabla 3.8</i>	<i>Opciones de Configuración de Paridad del Interfaz UART mediante el comando ATNB.....</i>	<i>94</i>
<i>Tabla 3.9</i>	<i>Opciones de Configuración de la Habilitación de Operación en Modo Transparente o Operación mediante la utilización de Tramas API, por medio del Comando ATAP.....</i>	<i>95</i>
<i>Tabla 3.10</i>	<i>Opciones de Configuración para el Establecimiento del Nivel de Potencia de Trasmisión mediante el Comando ATPL.....</i>	<i>95</i>
<i>Tabla 3.11</i>	<i>Opciones de Configuración de un Dispositivo Coordinador.....</i>	<i>97</i>
<i>Tabla 3.12</i>	<i>Opciones de Configuración de un Dispositivo Final.....</i>	<i>98</i>
<i>Tabla 3.13</i>	<i>Parámetros y Valores de Configuración del Dispositivo RFD ZigBee del Módulo de Temperatura y Humedad Relativa.....</i>	<i>100</i>
<i>Tabla 3.14</i>	<i>Consumo de Energía de los Dispositivos Principales del Módulo de Temperatura y Humedad Relativa.....</i>	<i>104</i>
<i>Tabla 3.15</i>	<i>Características y Valores Máximos de Operación Eléctricas del Microcontrolador ATmeg164P.....</i>	<i>111</i>
<i>Tabla 3.16</i>	<i>Asignación de Pines del Microcontrolador ATmega164P.....</i>	<i>116</i>
<i>Tabla 3.17</i>	<i>Parámetros y Valores de Configuración del Dispositivo RFD ZigBee del Módulo de Precipitación.....</i>	<i>126</i>
<i>Tabla 3.18</i>	<i>Asignación de Pines del Microcontrolador ATmega164P del Módulo de Precipitación.....</i>	<i>127</i>
<i>Tabla 3.19</i>	<i>Parámetros y Valores de Configuración del Módulo FFD ZigBee Coordinador del Módulo de Adquisición de Datos.....</i>	<i>132</i>
<i>Tabla 3.20</i>	<i>Configuración del Interfaz VISA para Comunicación Serial con el Módulo ZigBee Coordinador.....</i>	<i>136</i>
<i>Tabla 3.21</i>	<i>Construcción de una Tama API de Transmisión de 64-bits de Direccionamiento.....</i>	<i>142</i>
<i>Tabla 3.22</i>	<i>Cálculo del CheckSum para una Trama API Tx de 64-bits de Direccionamiento.....</i>	<i>143</i>
<i>Tabla 3.23</i>	<i>Presentación u Ocultación de Información de Temperatura, Humedad y Precipitación según Requerimientos del Usuario.....</i>	<i>159</i>

<i>Tabla 3.24</i>	<i>Sintaxis, Descripción y Respuesta de cada uno de los Comandos AT Utilizados para Iniciar la Comunicación GPRS y que describen a cada uno de los Estados de la Transmisión GPRS.....</i>	<i>167</i>
<i>Tabla 3.25</i>	<i>Función y Descripción de cada uno de los Estados de la Recepción GPRS TCP/IP.....</i>	<i>172</i>
<i>Tabla 4.1</i>	<i>Valores de Voltaje vs Temperatura del Sensor HMP45A.....</i>	<i>183</i>
<i>Tabla 4.2</i>	<i>Valores de Voltaje vs Humedad del Sensor HMP45A.....</i>	<i>185</i>
<i>Tabla 4.3</i>	<i>Resultados de Nivel de Señal RSSI en -dBm en una Topología Punto a Punto entre dos Dispositivos XBee-PRO.....</i>	<i>188</i>
<i>Tabla 4.4</i>	<i>Distribución de Frecuencia para los Datos de Error Absoluto de la Comparación de Valores de Temperatura.....</i>	<i>192</i>
<i>Tabla 4.5</i>	<i>Distribución de Frecuencia para los Datos de Error Absoluto de la Comparación de Valores de Humedad.....</i>	<i>193</i>
<i>Tabla 4.6</i>	<i>Especificaciones de Costos de Elementos y Dispositivos del Módulo de Temperatura y Humedad Relativa.....</i>	<i>196</i>
<i>Tabla 4.7</i>	<i>Especificaciones de Costos de Elementos y Dispositivos del Módulo de Precipitación.....</i>	<i>197</i>
<i>Tabla 4.8</i>	<i>Especificaciones de Costos de Elementos y Dispositivos del Módulo de Adquisición de Datos.....</i>	<i>198</i>
<i>Tabla 4.9</i>	<i>Especificaciones del Costo Total del Prototipo.....</i>	<i>199</i>

## RESUMEN

El presente proyecto tiene como fin, permitir al INAMHI contar con una estación automática cuyo monitoreo y adquisición de datos meteorológicos sea inalámbrico, utilizando los estándares IEEE 802.15.4 y “ZigBee”.

En este sentido, se implementa una red inalámbrica de área personal, en la banda ISM de 2.4 GHz, por medio de la instalación de módulos inalámbricos terminales en los puntos de medición remotos, tal que se procese y transmita información de los sensores de temperatura, humedad y precipitación, a un módulo coordinador. Este módulo es encargado de la administración y control de la red inalámbrica, así como la adquisición y tratamiento de la información recibida, para la correspondiente obtención, almacenamiento y presentación en tablas y gráficos de los valores de temperatura, humedad y precipitación. De igual manera, el módulo de adquisición de datos se encarga de enviar la información meteorológica a un servidor localizado en las oficinas del INAMHI en Quito, por medio de la utilización de la tecnología GPRS ya implementada y en funcionamiento en el INAMHI.

Los módulos terminales, han sido diseñados para ser localizados en sectores remotos donde se requiera obtener la información meteorológica. Por ende, necesitan ser acoplados a un sistema de energización DC que hace uso de paneles solares. El módulo coordinador de adquisición de datos, a diferencia de los módulos terminales, utiliza corriente alterna. Además, tanto los sensores como los módulos han sido sometidos a procesos de calibración que garantizan la medición y validez de la información obtenida.

Las pruebas de funcionamiento y transmisión han sido realizadas en campo, considerando un rango de distancia entre 10 m y 75 m especificado por el estándar IEEE 802.15.4.

## PRESENTACIÓN

El INAMHI no cuenta con estaciones automáticas inalámbricas, refiriéndose a las mismas como aquellas que realizan la adquisición de datos de las distintas variables meteorológicas de forma inalámbrica en los sectores remotos de medición, en los cuales se debe permitir la facilidad de movilidad de los distintos sensores, los mismos que puedan ser ubicados en puntos de medición adecuados. Se debe mencionar que el envío de la información por parte de la estación remota a la central del INAMHI en Quito, será realizado mediante la solución GPRS ya implementada y en funcionamiento.

Además, una de las funciones del INAMHI a través del departamento IDT, es la investigación e innovación tecnológica, razón por lo cual este prototipo pretende cumplir con la implementación de una estación automática propia, de bajo costo y diseñada en las instalaciones del INAMHI.

Al finalizar este proyecto, se tendrá una alternativa para la adquisición de datos meteorológicos de forma inalámbrica, con el fin de enviar los datos recolectados a través de la red GPRS y así monitorear una estación de forma remota.

# CAPÍTULO 1

## MARCO TEÓRICO

### 1.1. DESCRIPCIÓN DE LOS ESTÁNDARES DE COMUNICACIONES IEEE 802.15.4 Y ZIGBEE

En esta sección se describe de manera general a los estándares IEEE 802.15.4 y ZigBee, para tener nociones del funcionamiento de la red inalámbrica que el prototipo va utilizar.

#### 1.1.1. INTRODUCCIÓN AL ESTANDAR ZIGBEE [1]

El estándar ZigBee tiene como objetivo aplicaciones cuyos requerimientos sean de bajo consumo de energía, bajas velocidades de transmisión, bajo costo y larga vida de funcionamiento. Los dispositivos inalámbricos basados en el estándar ZigBee, operan en las bandas de frecuencia de 868MHz, 915MHz y 2.4GHz. La velocidad de transmisión máxima que se puede alcanzar es de 250Kbps.

ZigBee es un estándar desarrollado por la Alianza ZigBee, el cual tiene cientos de compañías miembros, de la industria de semiconductores y desarrolladores de software. El estándar ZigBee adoptó el IEEE 802.15.4 como protocolos de capa física (*Physical Layer PHY*) y control de acceso al medio (*Medium Access Control MAC*). Por ende, un dispositivo ZigBee es complementado con el estándar IEEE 802.15.4.

#### 1.1.1.1. Tipos de Redes Inalámbricas de Corto Alcance [2]

Se presentan dos tipos de redes inalámbricas de corto alcance:

- Redes Inalámbricas de Área Local (*Wireless Local Area Networks WLANs*).
- Redes Inalámbricas de Área Personal (*Wireless Personal Area Networks WPANs*).

Las redes inalámbricas de área personal (WPAN), presentan eficiencia de consumo de energía por parte de los dispositivos que operan dentro del espacio de área personal (*Personal Operating Space POS*), sin la necesidad de ninguna infraestructura. POS es definida como la región esférica que rodea un dispositivo inalámbrico y presenta un radio de 10 metros cuadrados.

Las redes inalámbricas de área personal se dividen a su vez en las siguientes categorías:

- WPANs de alta velocidad de transmisión (*High Rate HR WPANs*).
- WPANs de velocidad de transmisión media (*Medium Rate MR WPANs*).
- WPANs de baja velocidad de transmisión (*Low Rate LR WPANs*).

Un ejemplo de HR-WPANs es IEEE 802.15.3 con una velocidad de transmisión de 11 a 55 Mbps, el cual puede ser utilizado en aplicaciones inalámbricas de tiempo real como transmisión de video. Bluetooth, con una velocidad de transmisión de 1 a 3 Mbps, es un ejemplo de una red MR-WPAN. ZigBee, con una velocidad máxima de 250Kbps, es clasificado con una red de baja velocidad de transmisión LR-WPAN [3].

#### **1.1.1.2. Frecuencias de Operación y Velocidades de Transmisión [2]**

La última versión del estándar IEEE 802.15.4, presenta tres bandas de operación que se presentan a continuación:

- Banda de operación 868 – 868.6 MHz (868 MHz), la cual es utilizada en Europa.
- Banda de operación 902 – 928 MHz (915 MHz), la cual es parte de las bandas de frecuencias libres para investigación científica, industrial y médica (*Industrial, Scientific, and Medical ISM*), generalmente utilizada en Estados Unidos de América.
- Banda de operación 2400 – 2483.5 MHz (2.4 GHz), la cual es parte de las bandas de frecuencias ISM descritas anteriormente, es utilizada a nivel mundial.

La Tabla 1.1 provee detalles de cómo estas tres bandas de frecuencia son usadas en el estándar IEEE 802.15.4.

**Tabla 1.1 Velocidades de Transmisión del Estandar IEEE 802.15.4, Frecuencias de Operación, Modulación, Número de Canales y Métodos de Ensanchamiento del Espectro de Frecuencia**

Frecuencia (MHz)	Número de Canales	Modulación	Velocidad Chip (Kchip/s)	Velocidad Bit (Kb/s)	Velocidad de Símbolo (Ksymbol/s)	Método de Ensanchamiento de Frecuencia
868-868.6	1	BPSK	300	20	20	DSSS
902-928	10	BPSK	600	40	40	DSSS
2400-2483.5	16	O-QPSK	2000	250	62.5	Arreglo ortogonal de dimensión 16

Fuente: Elaboración propia basado en [2]

### 1.1.1.3. Tipos y Función de Dispositivos [2]

El estándar IEEE 802.15.4, establece dos tipos de dispositivos para una red inalámbrica:

- Dispositivos de Funcionamiento General (*Full – Function Devices* FFDs).
- Dispositivos de Funcionamiento Reducido (*Reduced – Function Devices* RFDs).

Un dispositivo de funcionamiento general puede realizar y cumplir cualquier rol en una red inalámbrica descrita por el estándar IEEE 802.15.4. Mientras que un dispositivo de funcionamiento reducido solo puede comunicarse con un dispositivo de funcionamiento total y se utiliza generalmente para aplicaciones simples. De esta forma los roles que cumple cada dispositivo se muestra en la Figura 1.1.



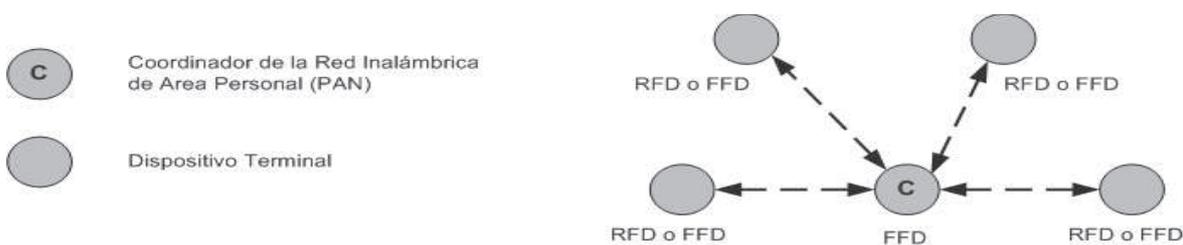
**Figura 1.1 Tipos y Roles de los Dispositivos ZigBee**

Fuente: Elaboración propia basado en [2]

#### 1.1.1.4. Topologías de Redes ZigBee [2]

El estándar IEEE 802.15.4 especifica dos tipos de topologías:

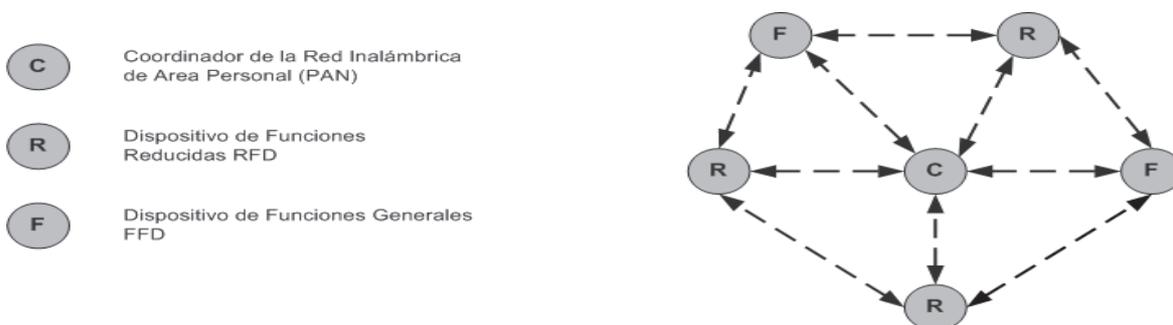
- Topología de Estrella, en la que un dispositivo de funcionamiento general (*Full-Function Device* FFD) presenta el rol de coordinador de la red inalámbrica de área personal (*Personal Area Network* PAN), encargado de establecer la red inalámbrica, para lo cual utiliza un identificador único, que no puede ser utilizado por ninguna red inalámbrica cercana.



**Figura 1.2 Topología tipo Estrella**

Fuente: [2]

- Topología tipo Malla, en la cual cada dispositivo establece un enlace de comunicación directo, siempre y cuando los dispositivos se encuentren lo suficientemente cercanos para establecer dicho enlace. En esta topología cualquier dispositivo FFD puede ser utilizado como Coordinador PAN, a diferencia de los dispositivos RFD, los cuales solo pueden comunicarse con un dispositivo Coordinador PAN o Router dentro de la red.



**Figura 1.3 Topología tipo Malla**

Fuente: [2]

Las funciones de un coordinador PAN, para control de la red inalámbrica son las siguientes:

- Establecer el tipo de direccionamiento (16 o 64 bits) para cada dispositivo en la red.
- Iniciar, entregar y enrutar los mensajes a través de la red.
- Seleccionar un identificador PAN único para la red. Este identificador permite a los dispositivos dentro de la red usar un direccionamiento corto de 16-bits y seguir permitiendo la comunicación con otros dispositivos a través de redes independientes.

Se debe notar que solo existe un coordinador PAN en cada red, el cual puede necesitar de periodos largos de activación, razón por la cual es preferible conectar este dispositivo a una fuente de energía principal y no a una batería.

### **1.1.2. CONCEPTOS DE COMUNICACIÓN ZIGBEE E IEEE 802.15.4**

Los conceptos básicos de comunicación ZigBee e IEEE 802.15.4, tienen como fin el de tener nociones para el entendimiento de la arquitectura ZigBee, que se revisará en secciones posteriores.

#### **1.1.2.1. Método de Acceso al Medio**

Se presentan dos métodos de acceso al medio.

##### *1.1.2.1.1. Método de Contención CSMA-CA [2]*

Este método de contención, utiliza el mecanismo de multi-acceso por medio de la detección de una señal portadora con anulación de colisiones (*Carrier Sense Multiaccess with Collision Avoid* CSMA-CA), el cual realiza un escaneo del canal de transmisión en busca de una señal portadora, para determinar si el canal de comunicación se encuentra libre u ocupado, en base a la medición del espectro de energía en el canal (*Energy Detection* ED) o detectando el tipo de señal que ocupa el mismo (*Carrier Sense* CS), previo a la transmisión de la información.

En el caso de la detección del espectro de energía (*Energy Detection* ED), el dispositivo actúa como receptor para detectar y estimar el nivel de señal en el canal deseado, tomando en cuenta que no se codifica la señal, ni se identifica el

tipo de señal que ocupa el canal; pero si se define un umbral de nivel de energía, que determina si el canal se encuentra libre u ocupado.

Por otro lado, se puede determinar el tipo señal o portadora (Carrier Sense CS) que se encuentra ocupando el canal, y en el caso de que se identifique una señal IEEE 802.15.4, el dispositivo puede considerar al canal ocupado, aun cuando el nivel de energía de la señal se encuentra por debajo del umbral previamente definido por el usuario.

Si el canal se encuentra libre, se reserva el ancho de banda del mismo, notificando por medio de señales de control al resto de dispositivos de la red, para garantizar que durante la transmisión de las tramas de información no existan colisiones. Sin embargo, pueden existir colisiones de tramas de control.

Cada vez que el canal se encuentre ocupado, se ejecuta un algoritmo de back-off estableciendo un tiempo de espera aleatorio antes de escuchar o censar nuevamente el canal de comunicación [4].

#### *1.1.2.1.2. Método de Contención Libre [2]*

En este método, el coordinador PAN define un espacio de tiempo (Guaranteed Time Slot GTS), que garantiza la transmisión de un dispositivo en particular. Para definir el espacio de tiempo de transmisión para cada uno de los dispositivos de la red, se debe asegurar que todos los dispositivos de la red se encuentren sincronizados, para lo cual se utiliza mensajes denominados "Beacon", los cuales presentan un formato específico para sincronizar todos los nodos de la red. La desventaja de utilizar este tipo de mensajes, es que generalmente los dispositivos de la red deben activarse únicamente para sincronizar sus relojes, lo cual determina que los dispositivos en una red Beacon consuma mayor cantidad de energía que en una red No Beacon.

#### **1.1.2.2. Métodos de Transferencia de Datos [2]**

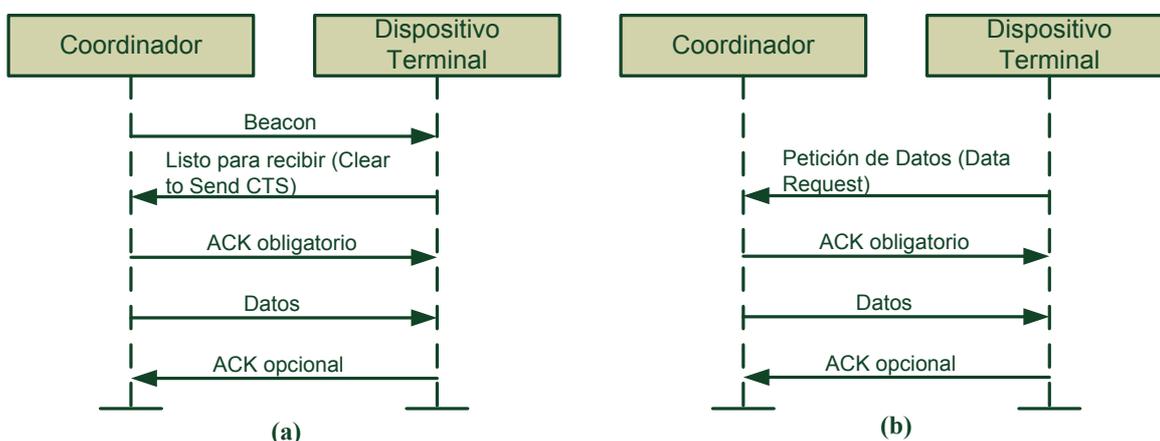
El estándar IEEE 802.15.4 presenta tres métodos de transferencia de datos: de un coordinador a un dispositivo terminal, de un dispositivo terminal a un

coordinador, y entre dispositivos terminales. Los tres métodos pueden ser utilizados en una topología tipo malla, sin embargo, solo el primer y segundo método puede utilizarse en una topología de estrella.

De igual manera, se debe considerar, si se utiliza un método de acceso al medio de contención o libre de contención, para la respectiva transferencia de datos.

#### 1.1.2.2.1. Transferencia de Datos de un Coordinador a un Dispositivo Terminal [2]

Si un dispositivo coordinador necesita enviar datos a un determinado dispositivo terminal en una red de contención libre, tal como se muestra en la Figura 1.4(a), se realiza una petición de envío de datos (*Request to Send RTS*) dentro de una trama Beacon. El dispositivo terminal a su vez, indica que se encuentra activo y listo para recibir información (*Clear to Send CTS*). El coordinador notifica que ha recibido el mensaje por medio de una señal de acuse de recibo (*Acknowledgment ACK*) mandatorio, para a continuación enviar los datos. El dispositivo terminal a su vez, envía un acuse de recibo (ACK) de que ha recibido los datos, sin embargo, este (ACK) es opcional.



**Figura 1.4** Transferencia de Datos desde un Coordinador a un Dispositivo Final: (a) Red Beacon y (b) Red No-Beacon

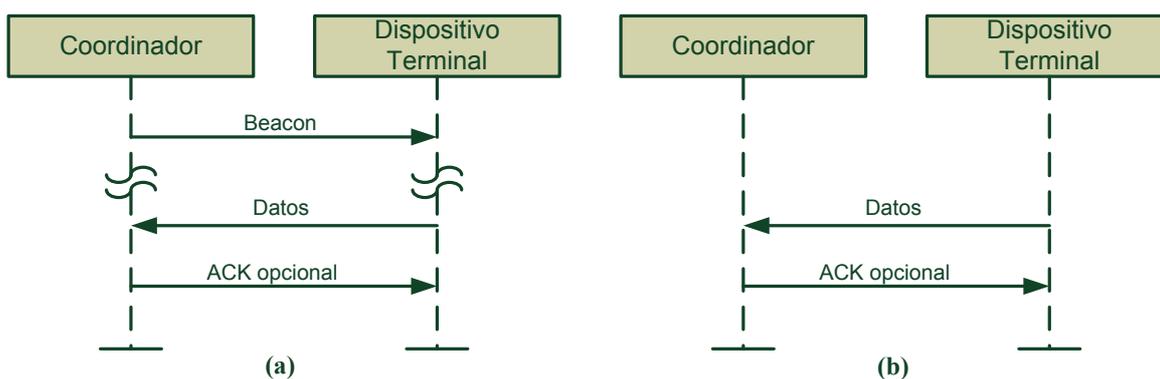
Fuente: [2]

Por otro lado, un dispositivo coordinador envía datos a un determinado dispositivo terminal en una red de contención, tal como se muestra en la Figura 1.4(b). El coordinador necesita esperar que el dispositivo terminal envíe un mensaje de petición de datos (*Data Request*). El dispositivo coordinador envía un mensaje de

acuse de recibo (ACK), para indicar que ha recibido el mensaje y a continuación enviar los datos. Finalmente el dispositivo terminal puede enviar un acuse de recibo (ACK), para indicar que ha recibido los datos, sin embargo, este paso es opcional.

#### 1.1.2.2.2. *Transferencia de Datos desde un Dispositivo Terminal a un Coordinador [2]*

Si un dispositivo terminal necesita enviar datos al dispositivo coordinador en una red de contención libre, el dispositivo coordinador debe inicialmente haber enviado una trama denominada Beacon, para sincronizar al dispositivo terminal, el cual a continuación puede enviar los datos. Finalmente si se requiere, el dispositivo coordinador envía un acuse de recibo (ACK), indicando que ha recibido los datos. Este proceso se puede observar en la Figura 1.5(a).



**Figura 1.5** Transferencia de Datos a un Coordinador en el Esquema IEEE 802.15.4: (a) Red Beacon y (b) Red No-Beacon

Fuente: [2]

Si un dispositivo terminal necesita enviar datos al dispositivo coordinador en una red de contención, el dispositivo terminal envía los datos tan pronto como el canal se encuentre libre. El acuse de recibo por parte del coordinador PAN es opcional. Este proceso se puede observar en la Figura 1.5(b).

#### 1.1.2.3. **Direccionamiento [2]**

En una red inalámbrica, cada uno de los dispositivos debe presentar una dirección única, que permita la comunicación entre dos o más dispositivos de la red. El estándar IEEE 802.15.4 establece dos tipos de direccionamiento:

- El direccionamiento reducido de 16-bits, permite acortar la longitud de los mensajes, y la comunicación dentro de una red inalámbrica. La combinación entre el identificador PAN y el direccionamiento reducido, permite la comunicación entre redes inalámbricas independientes.
- El direccionamiento completo de 64-bits, permite la utilización de 264 dispositivos dentro de una misma red inalámbrica.

#### **1.1.2.4. Asociación y Disociación de Dispositivos Terminales [4]**

ZigBee es considerado como una red inalámbrica Ad Hoc, por lo cual está conformado por un grupo de dispositivos inalámbricos que actúan dentro de un límite de acción propio, de tal forma que, un dispositivo puede pertenecer a una red inalámbrica por medio del servicio de asociación provisto por el estándar IEEE 802.15.4. De esta forma, un dispositivo envía un requerimiento de asociación al dispositivo coordinador de la red inalámbrica, el cual aceptará o negará este requerimiento. Por otro lado, la disociación es el proceso de notificación de un dispositivo terminal a un dispositivo coordinador, de su desvinculación de una red inalámbrica.

En un red Ad Hoc, la ruta de transmisión de un mensaje desde una fuente a un destino es seleccionado dinámicamente basado en la conectividad de la red, a diferencia de otras redes, en las cuales existe una infraestructura ya planteada y ciertos dispositivos actúan siempre como ruteadores.

## **1.2. ARQUITECTURA ZIGBEE**

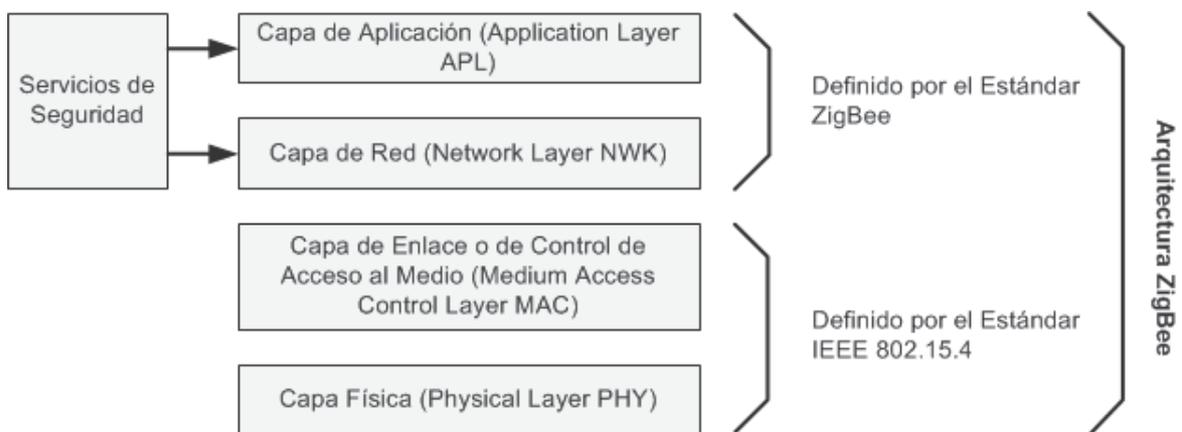
En esta sección se describe de manera general, cada una de las capas de la arquitectura ZigBee.

### **1.2.1. CAPAS DE LA ARQUITECTURA ZIGBEE E IEEE 802.15.4 [5]**

La arquitectura de una red ZigBee, se basa en el sistema abierto de interconexión (*Open System Interconnect* OSI) propuesto por la Organización de Estándares Internacionales (*International Standards Organization* ISO). En este sentido, se presentan dos capas inferiores física (*Physical* PHY) y de enlace o de control de

acceso al medio (*Medium Access Control MAC*), definidas por el estándar IEEE 802.15.4, y las capas de red (*Network NWK*) y de aplicación (*Application APL*) definidas por el estándar ZigBee. Se debe tomar en cuenta que las características de seguridad son definidas tanto por el estándar ZigBee e IEEE 802.15.4.

Entonces, se observa que tanto el estándar ZigBee como el estándar IEEE 802.15.4 se complementan para definir la arquitectura ZigBee, tal como se muestra en la Figura 1.6.



**Figura 1.6** Detalle de las Capas y Arquitectura de una Red ZigBee  
Fuente: [5]

La comunicación entre capas se realiza por medio de puntos de acceso de servicio (*Services Access Points SAPs*), los cuales permiten que el protocolo de una capa superior pueda requerir el servicio del protocolo de una capa adyacente inferior.

## 1.2.2. CAPA FÍSICA IEEE 802.15.4 (*PHYSICAL LAYER PHY*)

La capa física define características de hardware, funciones de protocolo y servicios brindados a la capa MAC. Esta capa es la más cercana al hardware y controla directamente la comunicación con el transmisor/receptor de radiofrecuencia.

### 1.2.2.1. Indicador de Calidad del Enlace [5]

El indicador de calidad del enlace (*Link Quality Indicator LQI*), muestra la calidad de los paquetes de datos recibidos por un receptor, para lo cual, se utiliza el nivel

de señal de recepción y la relación señal a ruido, como parámetros para cuantificar la calidad del enlace.

El nivel de señal de recepción (*Received Signal Strength* RSS) es una medida de la energía total de la señal de recepción. Mientras que, la relación señal a ruido (*Signal to Noise Ratio* SNR), en general considera una alta calidad de señal cuando el valor SNR es también alto, y por tanto con menor probabilidades de error en los paquetes recibidos.

Este indicador de calidad del enlace a más de ser usado por la capa de enlace o acceso al medio (MAC), puede también ser analizado por las capas de red (NWK) y aplicación (APL). La capa de red, hace uso del LQI, como uno de los factores para decidir qué camino usar para enrutar un mensaje. En consecuencia, mientras mayor sea el LQI, mayor será la probabilidad de que el mensaje llegue al destino.

#### **1.2.2.2. Asignación de un Canal Libre [5]**

La asignación de un canal de frecuencia libre (*Clear Channel Assessment* CCA), es un mecanismo que permite determinar si un canal se encuentra libre u ocupado, para lo cual fue previamente definido los parámetros de medición del espectro de energía (*Energy Detection* ED) y detección del tipo de señal que ocupa el canal de comunicación (*Carrier Sense* CS). De esta forma, se definen tres tipos de asignación de canal libre CCA:

- CCA modo 1: Hace uso de la detección de energía (ED), el cual considera un canal ocupado cuando el nivel (ED) es mayor que el umbral de nivel de energía previamente definido.
- CCA modo 2: Hace uso del resultado de la detección de portadora (CS), y considera el canal ocupado cuando la señal detectada es igual a las características de capa física del dispositivo inalámbrico.
- CCA modo 3: Este modo utiliza una combinación lógica (AND/OR) de los modos 1 y 2, y presenta las siguientes opciones:

- ✓ El nivel de energía detectado está por encima del umbral definido y la correspondiente portadora ha sido detectada.
- ✓ El nivel de energía detectada está por encima del umbral definido o la correspondiente portadora ha sido detectada.

### 1.2.2.3. Formato del Paquete de Datos de Capa Física PHY [5]

El formato de la unidad de datos del protocolo de capa física (*PHY Protocol Data Unit* PPDU), está conformada por los siguientes campos:

- Cabecera de sincronización (*Synchronization header* SHR).
- Cabecera de capa física (*PHY header* PHR).
- Carga útil de capa física (*PHY payload*).

Los campos antes mencionados se muestran en la Figura 1.7.



**Figura 1.7 Formato del Protocolo de la Unidad de Datos de Capa Física (PHY Protocol Data Unit PPDU)**

Fuente: [5]

La secuencia del preámbulo es utilizada por el receptor para obtener sincronización de símbolos y chips. Lo bits correspondientes a este sub-campo son ceros binarios excepto cuando se utiliza modulación de amplitud ASK. La longitud y duración del preámbulo de capa física para el caso de utilizar una frecuencia de operación de 2.4 GHz es de: 4 octetos, 8 símbolos y 128 us de duración.

El sub-campo del delimitador de inicio de trama (*Start-of-Frame Delimiter* SFD), indica el final del campo de la cabecera de sincronización (SHR), e indica el comienzo del campo de la cabecera de capa física (PHR). En el caso de utilizar

una frecuencia de operación de 2.4 GHz, este campo presenta una longitud de 8-bits o 2 símbolos, tal como se muestra en la Tabla 1.2.

**Tabla 1.2 Subcampo del Delimitador de Inicio (Start of Frame Delimitator SFD) con Excepción de las Tramas ACK de Capa Física**

Bits	0	1	2	3	4	5	6	7
Valores	1	1	1	0	0	1	0	1

Fuente: [5]

El sub-campo de longitud de trama, indica el número total de octetos en la carga útil o unidad de datos de capa física (*PHY Service Data Unit PSDU*). La longitud del PSD puede tener un valor entre 0 y 127 octetos. Basados en la unidad de datos del protocolo MAC (MPDU), el PSDU puede presentar longitudes variables de acuerdo a la Tabla 1.3.

**Tabla 1.3 Valores de Longitudes de Trama de Acuerdo a los Diferentes Tipos de Campos de Carga útil de Capa Física**

Valores de Longitud de Trama	Carga Útil (Payload) de Capa Física
0-4	Reservado
5	Acuse de recibo (ACK) MPDU
6-8	Reservado
9-aMaxPHYPacketSize	Cualquier otro MPDU

Fuente: [5]

Por tanto hay que tomar en cuenta que la longitud máxima de un paquete de capa física es de 127 octetos.

El último campo corresponde a la unidad de datos de capa física (*Packet Service Data Unit PSDU*), el cual contiene la trama de datos enviados por la capa de acceso al medio (MAC).

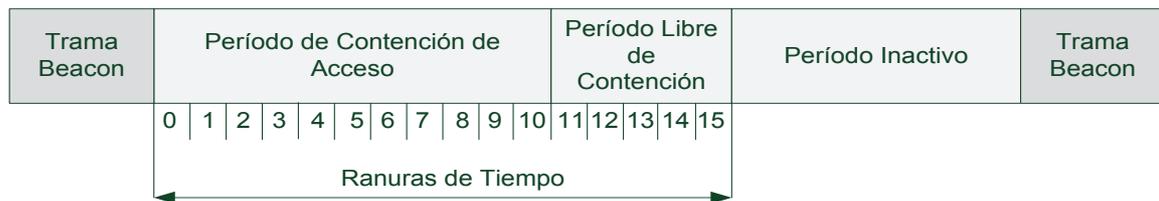
### **1.2.3. CAPA DE ACCESO AL MEDIO IEEE 802.15.4 (*MEDIUM ACCESS CONTROL MAC*)**

La capa de control de acceso al medio correspondiente a la capa 2 del modelo OSI, provee el interfaz entre la capa física y la capa de red. Esta capa es responsable de generar tramas denominadas *Beacons* utilizadas para sincronizar

los dispositivos en una red de contención libre. La capa MAC se encarga también de proveer servicios de asociación y disociación [2].

### 1.2.3.1. Tipos de Métodos de Acceso al Medio CSMA-CA [5]

El método de acceso al medio CSMA-CA puede ser clasificado como: CSMA-CA ranurado y CSMA-CA no ranurado. El método CSMA-CA ranurado es utilizado en redes de contención libre, y se refiere a la utilización de súper-tramas, en las cuales se definen tres periodos de tiempo clasificados como: período de contención de acceso (*Contention Access Period CAP*), período de libre contención (*Contention Free Period CFP*) y el período inactivo tal como se muestra en la Figura 1.8, los cuales permiten garantizar la transmisión por parte de un dispositivo en particular, tal como se describió en el método de contención libre. Además, se debe mencionar que una súper-trama se encuentra entre dos tramas *Beacon*, utilizadas para sincronización de los nodos de la red inalámbrica.



**Figura 1.8** Formatos de una Super-Trama

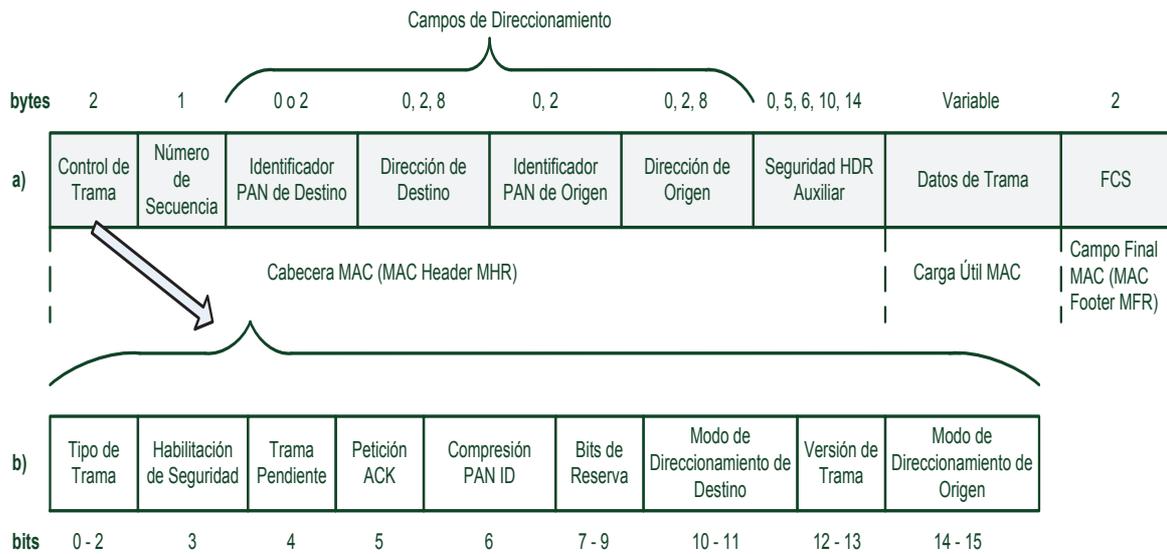
Fuente: [5]

Por otro lado, se hace uso del método CSMA-CA no ranurado, cuando no se utiliza súper-tramas en una red de contención. En su lugar se hace uso de tramas de datos MAC y de control MAC. Los métodos de acceso al medio fueron descritos en la sección referente a conceptos básicos de comunicación ZigBee.

### 1.2.3.2. Formato de Trama de la Capa de Acceso MAC [5]

La trama general de la capa de acceso MAC se muestra en la Figura 1.9, y está conformada por las siguientes secciones: cabecera (MAC Header MHR), carga útil (MAC Payload) y campo final (MAC Footer MFR).

El primer campo de control presenta 2 octetos, 16 bits o 2-bytes de longitud, y a su vez, este campo permite realizar las siguientes funciones:



**Figura 1.9 (a) Formato General de Trama MAC y (b) Detalles del Campo de Control**

Fuente: [5]

- Definir el tipo de trama mediante la utilización de bits 0-2. El estándar IEEE 802.15.4, especifica cuatro tipos de tramas de la capa de acceso MAC, las cuales son: Trama Beacon, trama de datos, trama del acuse de recibo (ACK), y la trama de comandos.
- Establecer una cabecera MAC auxiliar, en caso de que el bit 3 de seguridad se encuentre activado en 1L. Caso contrario, la cabecera MAC auxiliar tendrá 0 bits de longitud de campo.
- Determinar si el coordinador contiene datos pendientes a ser transmitidos a un determinado dispositivo, como parte del método de transmisión indirecta, cuando el bit número 4 se encuentra activado en 1L.
- Especificar si un dispositivo receptor, debe enviar un acuse de recibo (ACK) de la información recibida, mediante la activación de 1L de bit número 5.
- Comprimir la ID del coordinador PAN mediante la activación del bit número 6 en 1L, lo cual implica que solo se incluirá en la trama el identificador PAN de destino, de tal modo que el identificador de la fuente PAN, se asumirá que es el mismo que el de destino.
- Determinar el tipo de direccionamiento corto de 16 bits o ampliado de 64 bits del origen y destino, mediante la configuración de los bits 10-11 y 14-15 respectivamente.
- Conocer la versión de trama IEEE 802.15.4 que se está utilizando, mediante la configuración de los bits 12-13, ya que este estándar suele actualizarse.

El segundo campo correspondiente a la secuencia de numeración, presenta 1 octeto, 8-bits o 1-byte de longitud. Este campo es utilizado para distinguir entre tramas que hayan sido retransmitidas y por ende con la misma secuencia de numeración. La secuencia de numeración puede ser de dos tipos, secuencias de numeración *Beacon* (*Beacon Sequence Number* BSN) y secuencias de numeración de Datos (*Data Sequence Number* DSN).

Los campos de direccionamiento (*Addressing Fields*) corresponden al identificador PAN de destino, dirección de destino, identificador PAN de origen, y dirección de origen.

El campo auxiliar de seguridad de cabecera, es opcional y contiene información del nivel y tipos de claves de seguridad, que se utilizan para asegurar la trama de la capa de acceso MAC. Este campo puede presentar de 0 a 14 octetos (bytes) de longitud.

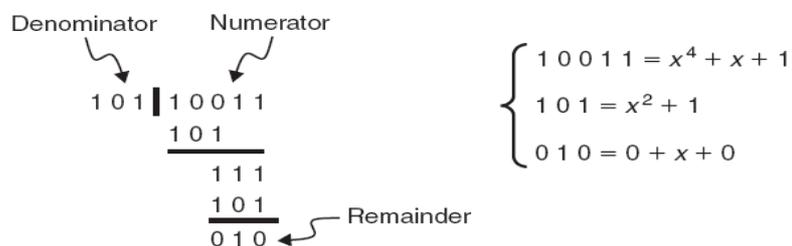
La sección final de una trama MAC, contiene el campo de secuencia de chequeo de trama (*Frame Check Sequence* FCS) de 16-bits o 2-bytes basado en el chequeo de redundancia cíclica (*Cyclic Redundancy Check* CRC), que permite detectar errores en el paquete de datos de la trama de recepción.

En este sentido, en el dispositivo de transmisión, todos los bits correspondientes al campo de cabecera de trama MAC (*MAC Header* MHR) y del campo de datos de carga útil (*MAC payload*), son tratados como coeficientes de un polinomio, el cual es dividido para otro polinomio conocido tanto por el transmisor como por el receptor. El resultado de dicha división se denomina secuencia de chequeo de trama (*Frame Check Sequence* FCS), el cual se añade al campo final de la trama de transmisión (*MAC Footer* MFR). El dispositivo de recepción realizará la misma división en expectativa de obtener el mismo resultado. En el caso de que el cociente calculado en recepción, no sea el mismo que el cociente provisto y obtenido del campo final de trama MFR de transmisión, el dispositivo de recepción concluirá que la trama recibida presenta errores. Un ejemplo se muestra en la Figura 1.10.

El FCS en el estándar IEEE 802.15.4 puede ser generado siguiendo los siguientes pasos:

- a) Definir el polinomio  $M(x)$ , que representará a los bits totales de las secciones de cabecera y carga útil de la trama MAC.
- b) Multiplicar  $M(x)$  por  $x^{16}$
- c) Dividir el polinomio  $M(x)$  por  $x^{16}$  para el siguiente polinomio:

$$G_{16}(x) = x^{16} + x^{12} + x^5 + 1$$



**Figura 1.10** Ejemplo de una División Simple de Módulo 2  
 Fuente: [5].

1.2.3.2.1. Trama de Datos [5]

La trama de datos de la capa de acceso (MAC), contiene información de la capa de red (NWK) dentro del campo de carga útil, denominada como una unidad de datos de servicio de la capa de acceso MAC (MAC Service Data Unit MSDU). En la Figura 1.11 se muestra la trama mencionada.

bytes	2	1	Variable	0, 5, 6, 10, 14	Variable	2
	Control de Trama	Número de Secuencia	Campos de Direccionamiento	Seguridad HDR Auxiliar	Datos de Trama	FCS
	Cabecera MAC (MAC Header MHR)				Carga Útil MAC	Campo Final MAC (MAC Footer MFR)

**Figura 1.11** Formato Trama de Datos de la Capa MAC  
 Fuente: [5]

1.2.3.2.2. Trama de Acuse de Recibo (ACK) [5]

La trama de acuse de recibo (Acknowledgment ACK) se utiliza para verificar si un dispositivo receptor o transmisor ha recibido el paquete de datos exitosamente. La Figura 1.12 muestra la trama en cuestión.



**Figura 1.12 Formato de Trama de Acuse de Recibo (ACK)**

*Fuente: [5]*

#### 1.2.4. CAPA DE RED ZIGBEE NWK (*NETWORK LAYER*) [5]

La capa de red establece el interfaz entre las capas de acceso (MAC) y aplicación (APL) y es responsable del manejo de la formación de red y ruteo, lo cual implica el seleccionamiento del mejor camino a través del cual el mensaje será enviado a los correspondientes dispositivos de destino. El coordinador ZigBee y los ruteadores son responsables del descubrimiento y del establecimiento de las rutas en la red. Se debe mencionar que un dispositivo terminal no puede realizar el descubrimiento y establecimientos de rutas. La capa de red (Network NWK) del dispositivo coordinador, es responsable del establecimiento de una nueva red y de seleccionar la topología de red (árbol, estrella o malla). De igual manera el coordinador ZigBee se encarga de asignar las direcciones de red a los dispositivos dentro de su red [2].

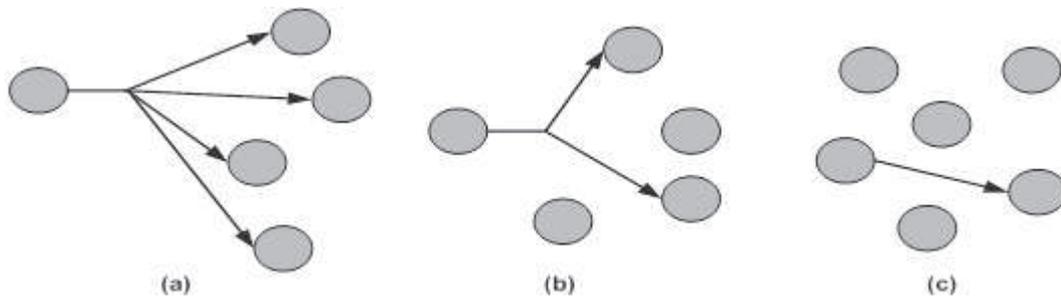
La capa de red limita la distancia que un paquete de datos puede recorrer dentro de una red. La distancia es definida como el número de saltos que realiza el paquete, determinado por un parámetro denominado radius. Por ejemplo, si el parámetro radius presenta un valor inicial de 3, esto significa que el paquete no será transmitido más de tres veces, determinando que en cada uno de los saltos el parámetro radius disminuirá en 1 hasta llegar a cero, punto en el cual, el paquete no será retransmitido.

El mecanismo de comunicación puede ser dividido en tres categorías: difusión, multipunto y punto a punto, tal como se muestra en la Figura 1.13.

##### 1.2.4.1. Redes ZigBee de Difusión [5]

En una red de difusión, un mensaje es recibido por todos los dispositivos dentro de una misma frecuencia o canal de comunicación, a pesar de que la dirección de

origen o identificador PAN sean diferentes. La difusión de una trama en una red IEEE 802.15.4, se realiza por medio de un direccionamiento corto de 16-bits (0xffff).



**Figura 1.13** Redes de: (a) Difusión, (b) Punta a Multipunto y (c) Punto a punto  
Fuente: [5]

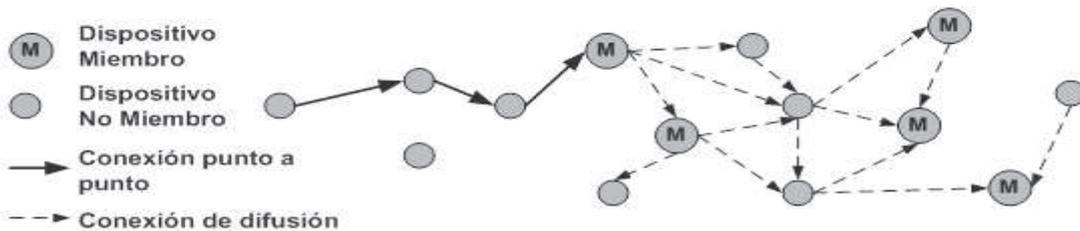
En una red con una gran cantidad de dispositivos, se dificulta que cada uno de ellos envíe un acuse de recibo (ACK), por lo cual, se utiliza un mecanismo de acuse de recibo pasivo, donde un dispositivo funciona como receptor una vez que haya enviado un mensaje de difusión, a la espera de que cualquiera de los dispositivos vecinos re-difundan la misma trama. Una trama re-difundida es la indicación de que el dispositivo vecino a recibido y retransmitido el mensaje de difusión exitosamente. En una red de difusión, los dispositivos terminales no utilizan acuse de recibo ACK.

#### 1.2.4.2. Redes ZigBee Punto a Multipunto [5]

El objetivo dentro de una red punto-multipunto, es el de enviar mensajes a un grupo de dispositivos dentro de la red. En este tipo de redes un dispositivo puede ser parte de un grupo de difusión múltiple, en cuyo caso la difusión del mensaje se realizará únicamente a los miembros del mismo grupo. Por otro lado, un dispositivo puede enviar un mensaje a un grupo de dispositivos, sin ser parte del grupo de difusión múltiple, mediante el enrutamiento del mensaje a un solo miembro del grupo mencionado, de tal manera que, sea éste quien difunda el mensaje al resto de dispositivos del grupo, tal como se muestra en la Figura 1.14.

En este tipo de redes, se puede limitar el número de veces que un paquete de difusión puede ser re-transmitido. De igual manera se debe tomar en cuenta que

una red punto-multipunto, es usada para la transmisión de tramas de datos y no de comandos.



**Figura 1.14** Enrutamiento Punto a Multipunto iniciado por un Dispositivo que es Miembro de un Grupo de Difusión Múltiple

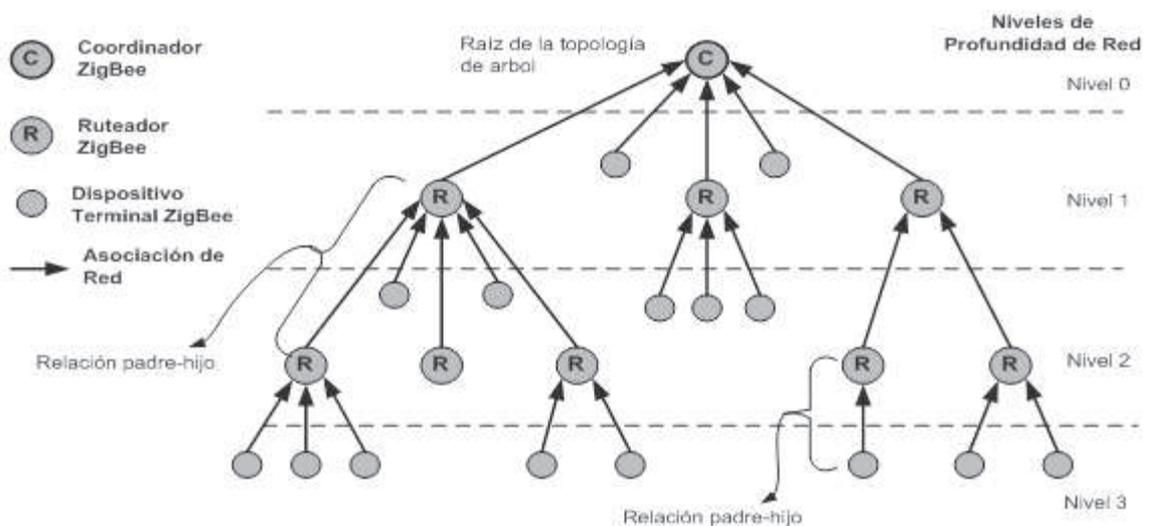
Fuente: [5]

### 1.2.4.3. Topologías de Red [5]

En la arquitectura ZigBee se presenta dos tipos de topologías que pueden ser utilizadas de acuerdo a los requerimientos del usuario. Estas topologías son: topología de árbol y topología de estrella.

#### 1.2.4.3.1. Topología de Árbol [5]

Este tipo de topología es jerárquica y presenta al coordinador o router ZigBee como un dispositivo padre que tienen bajo su control y manejo a dispositivos hijos, tal como se muestra en la Figura 1.15. Un dispositivo padre tiene la facultad de aceptar o denegar la asociación de otro dispositivo y de igual forma se encarga de enrutar mensajes de los dispositivos hijos.



**Figura 1.15** Relación entre Dispositivos Padres e Hijos en una Topología de Árbol

Fuente: [5]

La métrica de una red ZigBee se define como el mínimo de saltos requeridos para que una trama llegue al coordinador, en el caso de utilizar enlaces de dispositivos padre-hijos. Cuando el coordinador se encuentra directamente enlazado a un dispositivo hijo, se dice que la métrica de red es de 1, ya que entre ambos dispositivos existe un solo salto.

El estándar ZigBee define un mecanismo para asignación de direcciones de los dispositivos dentro de una topología de árbol. Este mecanismo se denomina asignación de direcciones distribuidas por defecto, en donde el dispositivo coordinador ZigBee proporciona a cada dispositivo padre, un sub-bloque de direcciones de red, de tal forma que un dispositivo padre pueda a su vez, asignar una dirección a cada uno de los dispositivos hijos bajo su control. El coordinador ZigBee tiene la potestad de determinar el número máximo de dispositivos hijos que cada uno de los dispositivos padres pueden asociar.

El mecanismo de asignación de direcciones distribuidas por defecto, utiliza parámetros como la métrica de red y el máximo número de dispositivos hijos, para determinar la asignación de direcciones.

#### *1.2.4.3.2. Topología de Estrella [5]*

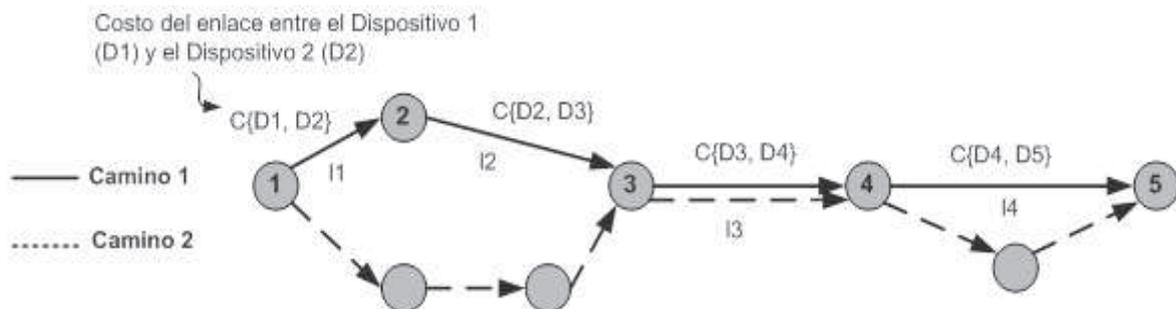
En este tipo de topología, a diferencia de la topología de árbol, no se establece jerarquía alguna, y la ruta desde un dispositivo origen a un dispositivo destino se crea bajo demanda y puede ser modificada de forma dinámica creando nuevas rutas, en caso de que la ruta previamente utilizada se encuentre des-habilitada.

#### **1.2.4.4. Ruteo [5]**

El proceso de ruteo consiste en seleccionar el camino a través del cual el mensaje será retransmitido al dispositivo de destino. Se debe mencionar que un dispositivo final no puede realizar el seleccionamiento de ruta, este proceso puede ser únicamente realizado por dispositivos coordinadores o ruteadores ZigBee.

Para el seleccionamiento de la mejor ruta, se debe tomar en cuenta parámetros como la longitud de la ruta (L), que indica el número de dispositivos en la ruta

desde un dispositivo origen a un dispositivo destino, la calidad de enlace y el número de saltos. En consecuencia, se define el costo de enlace, el cual muestra la probabilidad de que un paquete sea entregado exitosamente. El costo de enlace presenta un valor para cada enlace de la ruta ( $C[D_i, D_{i+1}]$ ), mostrando que mientras menor es la probabilidad de éxito, mayor es el costo del enlace. En la Figura 1.16, se observa la selección de la mejor ruta desde un dispositivo origen a un dispositivo destino.



**Figura 1.16 Análisis del Costo del Enlace**  
Fuente: [5]

Los dispositivos coordinador o ruteador ZigBee encomendados del proceso de seleccionamiento de la mejor ruta, manejan y administran tablas, las cuales presentan las siguientes descripciones:

- Tabla de enrutamiento, usada para determinar el próximo salto de una ruta, haciendo uso de cuantificaciones como el estatus de una ruta y la capacidad de enrutamiento de un dispositivo.
- Tabla de selección de ruta, la cual contiene los costos de enlaces, la dirección de un dispositivo de origen, y la dirección del último dispositivo que realizó retransmisión de la petición de ruta al dispositivo en cuestión, de tal forma que se envíe la ruta seleccionada al dispositivo de origen. El contenido de esta tabla es temporal a diferencia de la tabla de enrutamiento mencionada en el anterior punto.
- Tabla de dispositivos vecinos, utilizada para obtener información de dispositivos cercanos dentro del rango de transmisión. Esta tabla se actualiza cada vez que se reciba un paquete de dispositivos vecinos, y puede ser utilizada en el caso de buscar un nuevo dispositivo padre.

### **1.2.5. CAPA DE APLICACIÓN APL (*APPLICATION LAYER*)**

La capa de aplicación (*Application Layer APL*), presenta el protocolo más alto en una red inalámbrica ZigBee y es el entorno en el que los objetos de aplicación se organizan para controlar y gestionar las capas y protocolos en un dispositivo ZigBee. Los objetos de aplicación permiten personalizar a los dispositivos para diversas aplicaciones. En un solo dispositivo se puede encontrar hasta 240 aplicaciones [2].

La recepción y transmisión de datos entre objetos de aplicación se realiza por medio de los puntos de acceso. Cada aplicación presenta una única dirección de punto final numeradas desde el 1 al 240. La dirección 0 es reservada para uso del objeto de dispositivo ZigBee (*ZigBee Device Object ZDO*). La configuración del punto final de direccionamiento en 255, implica la difusión de un mensaje a todos los objetos de aplicación [5].

Los objetos de dispositivos ZigBee (*ZigBee Device Objects ZDO*) establecen un interfaz entre la subcapa de soporte de aplicación y el marco de aplicación. El ZDO, permite la configuración de los dispositivos ZigBee como coordinador, router o dispositivo final [5].

## **1.3. DESCRIPCIÓN GENERAL DE LAS TECNOLOGÍAS GSM Y GPRS**

En esta sección se describe de manera general las características principales del sistema global de comunicaciones móviles (*Global System for Mobile Communications GSM*) y el funcionamiento del servicio de paquetes de radio (*General Packet Radio Service GPRS*), para contar con nociones básicas, que permitan la implementación de la transmisión GPRS punto a punto.

### **1.3.1. DESCRIPCIÓN GENERAL DEL SISTEMA GLOBAL DE COMUNICACIÓN MÓVIL (*GLOBAL SYSTEM FOR MOBILE COMMUNICATION GSM*) [6]**

Algunas de las características importantes que describen al sistema global de comunicación móvil GSM, se presentan a continuación:

- Es un estándar abierto que permite acceso a sus especificaciones.
- Los interfaces de la arquitectura GSM se encuentran estandarizadas, lo que permite la participación de varios fabricantes.
- Establece seguridades por medio de una transmisión encriptado.
- Permite el manejo eficiente de suscriptores mediante la utilización del módulo de identificación de suscriptor (*Subscriber Identity Module* SIM).
- Utilización de los métodos FDMA y TDMA, para optimizar el uso del espectro de frecuencias.
- Requerimientos de bajo consumo de energía.
- Señalización de servicios y gran calidad de voz, por medio de transmisión digital.
- Permite la adaptación de actualizaciones.

#### **1.3.1.1. Interfaz de Aire GSM [6]**

El interfaz de aire de movilidad ilimitada (*Unlimited Mobility* UM), es utilizado tanto por GSM como por GPRS. Los bloques de frecuencia de subida (*Uplink* UL) y de bajada (*Downlink* DL) son divididos en bandas de frecuencia de 200 KHz de ancho de banda por medio del método de multiacceso de división de frecuencia (FDMA). A su vez, cada ancho de banda es dividido en 8 ranuras de tiempo utilizando el método de multiacceso de división de tiempo (TDMA), lo que permite que 8 usuarios utilicen el mismo ancho de banda simultáneamente. A las ocho ranuras de tiempo se las denomina trama TDMA.

Cuando se establece un canal de tráfico para una determinada estación móvil (*Mobile Station* MS), se le asigna una banda de frecuencia determinada y un número de ranura de tiempo UL/DL. De esta manera, una estación móvil MS puede transmitir y recibir mensajes de señalización, datos o señales voz digitalizada. En el caso del tratamiento de señales de voz analógica, se debe previamente digitalizar para luego codificar el canal (compresión y adicionar redundancia), antes de ser transmitido.

### 1.3.2. OBJETIVOS, VENTAJAS Y CONCEPTOS GENERALES DEL SERVICIO DE PAQUETES DE RADIO (*GENERAL PACKET RADIO SERVICE GPRS*) [6]

Los objetivos y ventajas de GPRS se dan en función de los requerimientos de usuarios (*User U*) y de los proveedores de la red de acceso (*Network Providers NP*). A continuación se presenta algunos de los objetivos y ventajas de GPRS..

- Permitir el acceso a redes LAN.
- Proveer velocidades de transmisión razonablemente altos (U).
- Permitir el acceso a la red en todo momento a un suscriptor que requiere de servicios de voz y datos (U).
- Ofrecer un acceso flexible, ya sea para muchos suscriptores a la vez con baja velocidad de datos o para pocos suscriptores con altas velocidades de datos (NP).
- Ofrecer acceso de bajo costo de nuevos servicios (U+NP).

La mayor ventaja de GPRS, es el uso de la conmutación de paquetes, lo cual implica las siguientes características:

- Los datos pueden ser directamente compartidos con compañías de internet o intranet.
- Los paquetes de un usuario pueden ser transmitidos vía varios ranuras de tiempo del interfaz de aire.
- Las ranuras de tiempo no son usadas continuamente y pueden ser compartidas entre usuarios.
- Los recursos son asignados a un determinado usuario solo lo necesite.
- GPRS es implementado dentro del estándar GSM, por lo que se hace necesario la utilización de nuevas frecuencias.

Se debe notar que las redes GSM que utilizan conmutación de circuitos, tuvieron que acoplarse a la conmutación de paquetes para implementar GPRS. Algunas de las aplicaciones y alcances de la red GPRS se puede observar en la Figura 1.17.



**Figura 1.17** Aplicaciones y Alcances GPRS

Fuente: [6]

### 1.3.2.1. Manejo del Recurso de Radio-Frecuencia en GPRS.

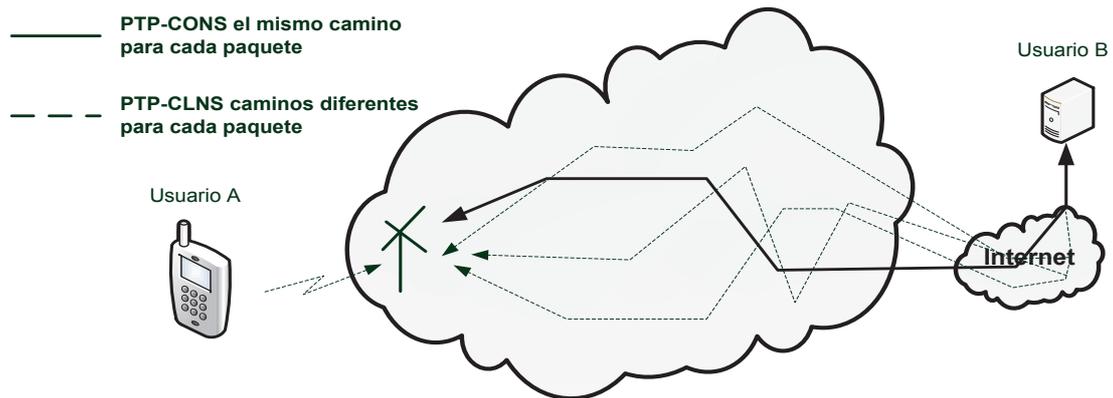
En GSM un canal de usuario es definido por un número de ranura de tiempo y par de frecuencias (UL y DL). Por otro lado, en GPRS la asignación de portadoras y números de ranuras de tiempo para un usuario es manejado de forma dinámica y es aplicado tanto para UL y DL. La asignación de recursos para UL y DL es manejado de forma separada, lo cual significa que mientras un suscriptor hace uso de una ranura de tiempo DL particular, otro usuario puede estar usando la ranura de tiempo UL correspondiente. Esto se realiza mediante las funciones de gestión de recursos de radio [6, pp. 33-34].

### 1.3.2.2. Servicios de Red [6]

En GPRS se define dos tipos de servicios de red los cuales se presentan en la Figura 1.18 y se definen como:

- Servicios de red punto-a-punto sin conexión (*Point-to-Point Connectionless Network Service PTP-CLNS*).
- Servicios de red punto-a-punto orientado a la conexión (*Point-to-Point Connection Orientated Network Service PTP-CONS*).

En el servicio de red PTP, los datos pueden ser transferidos entre usuarios o entre un usuario y un servidor, de los cuales uno debe realizar la función de suscriptor GPRS.



**Figura 1.18 Servicio de Transmisión GPRS**

Fuente: [6]

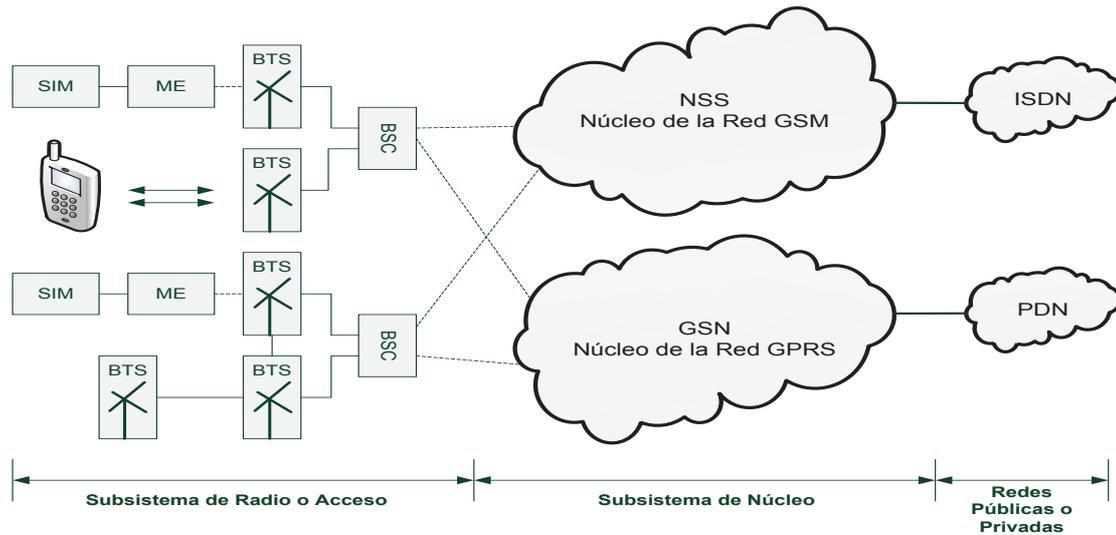
En PTP-CLNS, cada paquete de datos es enrutado de forma diferente a los demás paquetes de datos a través de la red. Por tanto, ninguna conexión es necesaria establecerla de antemano para que cada paquete pueda encontrar su propio camino a través de la red. Este servicio también llamado de datagrama, es útil en redes de gran tráfico o congestión de datos.

El PTP-CONS es orientado a la conexión, lo cual significa que un camino dedicado se establece entre dos usuarios antes de comenzar la transmisión.

### 1.3.3. ARQUITECTURA GPRS [6]

La red GPRS puede ser dividida en dos subsistemas: acceso y núcleo. Por tal razón, se debe considerar que GPRS se establece dentro de una red de conmutación y enrutamiento orientado al manejo de tráfico de paquetes, la cual es definida como el nodo de soporte de datos GPRS (*GPRS Support Node for Data GSN*). Por otro lado, GSM se implementa dentro de una red de conmutación de circuitos, conocida como el subsistema de la red de conmutación (*Network Switching Subsystem NSS*). NSS y GSN conforman el núcleo de una red que integra tanto GSM como GPRS, y que además, se encuentra separado del subsistema de radio o de acceso, como se observa en la Figura 1.19.

El subsistema de radio (*Radio Subsystem RSS*), está conformado por el conjunto de estaciones base de transmisión-recepción, dispositivos móviles y el interfaz de aire UM. A la red de estaciones bases de transmisión-recepción se lo conoce como el subsistema de estaciones base (*Base Station Subsystem BSS*).



**Figura 1.19 Separación Física de la Sección de Acceso de la Red del Núcleo GSM y GPRS**

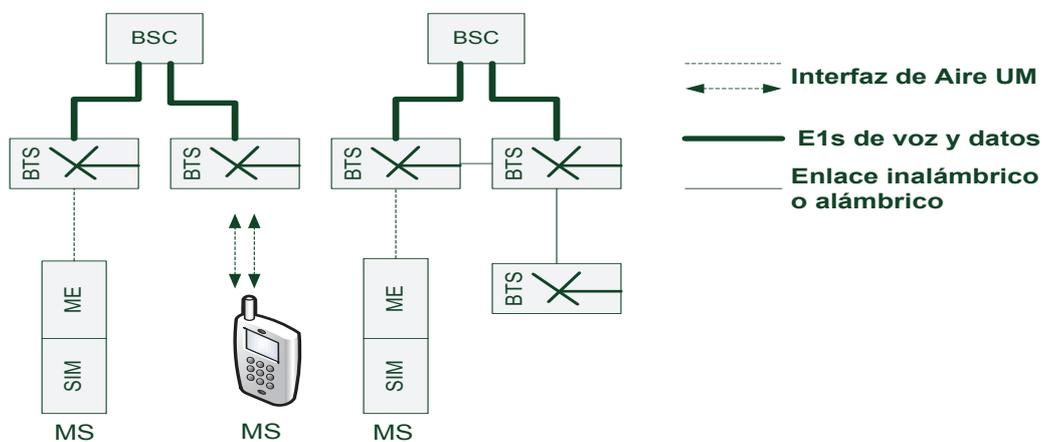
Fuente: [7]

### 1.3.3.1. Subsistema de Radio o de Acceso [6]

Este subsistema se muestra en la Figura 1.20, el cual se encuentra conformado por los siguientes elementos:

- Estación móvil (*Movil Station MS*).
- Estación Base de Transmisión-Recepción (*Base Transceiver Station BTS*).
- Estación Base de Control (*Base Station Controller BSC*).

Se debe mencionar que el subsistema de radio es el mismo tanto para GSM como para GPRS, con excepción de algunos elementos GPRS que se mencionan en la sección 1.3.3.2.

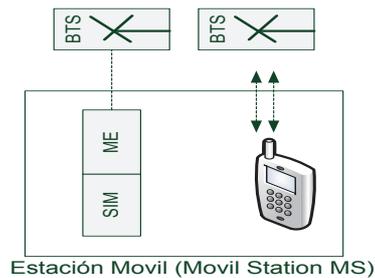


**Figura 1.20 Subsistema de Acceso**

Fuente: [6]

### 1.3.3.1.1. Estación Móvil (Movil Station MS) [6]

Es el dispositivo terminal o final que hace uso un usuario. Este dispositivo se encuentra conformado a su vez por el equipo móvil (*Mobile Equipment ME*) y por el módulo de identidad de suscriptor (*Subscriber Identity Module SIM*), como se muestra en la Figura 1.21.



**Figura 1.21 Estación Móvil GSM/GPRS.**

Fuente: [6]

El equipo móvil presenta las siguientes funciones:

- Interfaz de usuario.
- Entrada y salida de señales de voz.
- Manejo de datos recibidos vía GPRS.
- Marcado y almacenamiento de números.
- Funciones avanzadas de menú.
- Transmisor-receptor desde y hacia la BTS.

El módulo de identidad del suscriptor realiza funciones como:

- Autenticación dentro de la red GPRS.
- Establecer un algoritmo de cifrado, utilizado en cada conexión individual para evitar que la comunicación sea percibida de forma ilegal.
- Identificación de usuario.
- Contener una memoria adicional para agenda del usuario.

### 1.3.3.1.2. Estación Base de Transmisión-Recepción (Base Transceiver Station BTS) [6]

Una BTS permite las comunicaciones entre una estación móvil MS y la red GSM/GPRS. Además, implementa una celda de cobertura de tamaño limitado. La

BTS a más de presentar un interfaz de aire UM para comunicarse con una estación móvil MS, establece un interfaz para comunicarse con estación base de control. Las funciones de una BTS, se presentan a continuación:

- Transmisión de los datos cifrados y comprimidos.
- Modulación
- Control de la calidad de la transmisión.
- Control de potencia de transmisión y cobertura.

#### *1.3.3.1.3. Estación Base de Control (Base Station Controller BSC) [6]*

Una BSC se encarga del manejo y control de las BTSs, y de la gestión de las estaciones móviles activas en las correspondientes celdas. Algunas de las funciones de la BSC, se presentan a continuación:

- Tareas de conmutación simples.
- Control de las BTSs.
- Señalización.
- Punto de acceso del operador.

#### **1.3.3.2. Elementos GPRS en el Subsistema de Radio o Acceso [7]**

En una red GPRS, las entidades necesarias dentro del subsistema de acceso son:

- Unidades de Codificación de Canales (*Channel Codec Units* CCU).
- Unidades de Control de Paquetes (*Packet Control Units* PCU).

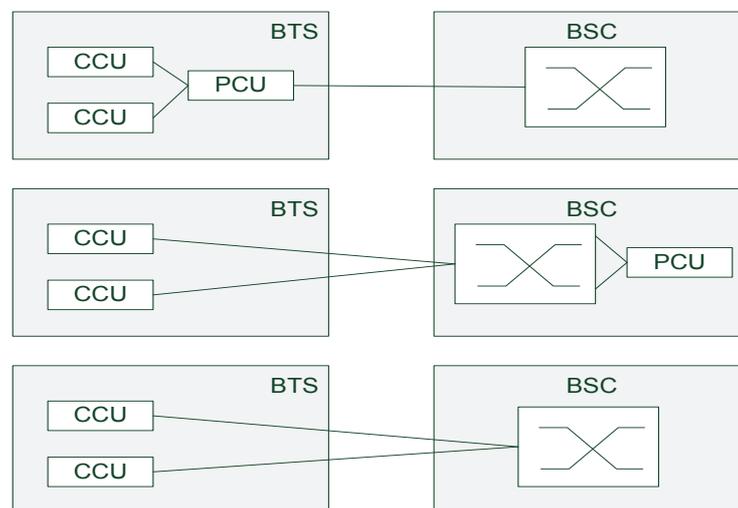
El CCU implementa funciones de software dentro de cada BTS, mientras que el PCU implementa funciones de hardware en la BTS y BSC, tal como se observa en la Figura 1.22.

Las funciones del CCU son:

- Codificación del canal usando, incluyendo corrección de errores FEC (*Forward Error Correction*) e intercalado (*interleaving*) y multiplexación.
- Funciones de medición de canales de radiofrecuencia, incluyendo calidad del nivel de recepción (RxQual), nivel de señal de recepción (RxLev), y valores relacionados a avances de temporización (*Timing Advance TA*).

El PCU es responsable de las siguientes funciones:

- Funciones de gestión del canal de radiofrecuencia, incluyendo control de potencia o energía (*Power Control PC*), control de congestión, y difusión de información de control.
- Segmentación del PDU de la capa LLC (transmisión del enlace de bajada DL).
- Re-ensamblaje del PDU de la capa LLC (transmisión del enlace de subida UL).
- Funciones para la transmisión de datos UL y DL.
- Funciones ARQ (*Automatic Repeat Request*).
- Funciones de control de acceso al canal, incluyendo peticiones de acceso al canal.



**Figura 1.22** Esquema del Subsistema de Acceso GPRS.

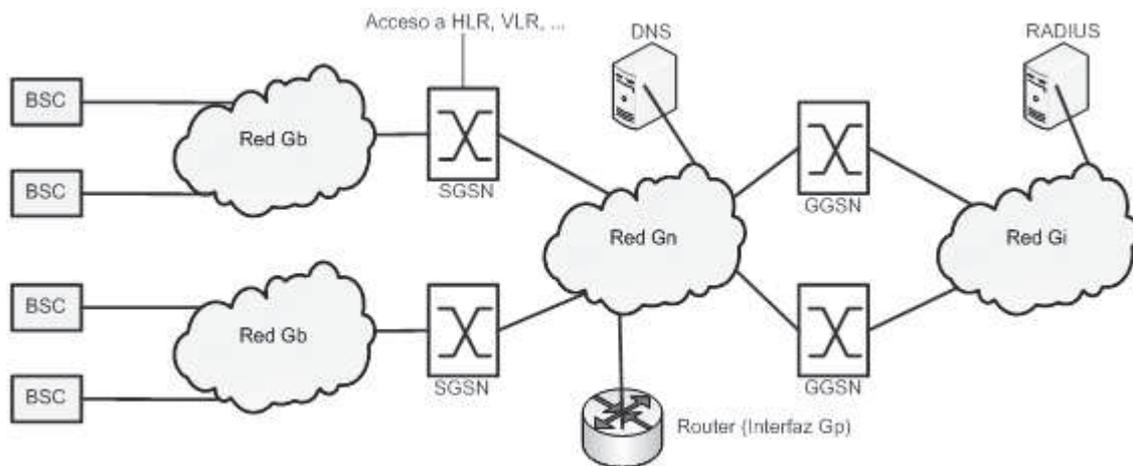
Fuente: [7]

### 1.3.3.3. Subsistema de Núcleo de una Red GPRS [7]

El núcleo de una red GPRS tiene como función interconectar nodos, routers, servidores, a través de interfaces definidas, para soportar tráfico orientado a paquetes de datos, basado en protocolos IP, tales como e-mail, HTTP, FTP, etc.

Este subsistema se muestra en la Figura 1.23, y está conformado por los siguientes elementos:

- Nodo de soporte de servicio GPRS (*Serving GPRS Support Node* SGSN).
- Nodo de soporte de puerta de acceso GPRS (*Gateway GPRS Support Node* GGSN).
- Red de acceso para interconectar las PCUs y las SGSNs por medio del interfaz Gb.
- Red de núcleo para interconectar las SGSNs y GGSNs por medio del interfaz Gn.
- Elementos adicionales como servidores de autenticación RADIUS y DNS conectados a través del interfaz Gi.



**Figura 1.23 Componentes del Núcleo de una Red GPRS**

Fuente: [7]

#### 1.3.3.3.1. *Nodo de Soporte de Servicio GPRS (Serving GPRS Support Node SGSN) [7]*

El nodo de soporte de servicio GPRS SGSN, permite conectar el subsistema de radio o acceso con el subsistema de núcleo. El SGSN presenta un registro de locación SGSN (*SGSN Location Register* SLR), el cual almacena información de los suscriptores. Las funciones que el SGSN realiza se muestran a continuación:

- Funciones de gestión de movilidad como: conexión, desconexión, actualización del área de enrutamiento y funciones de paging.

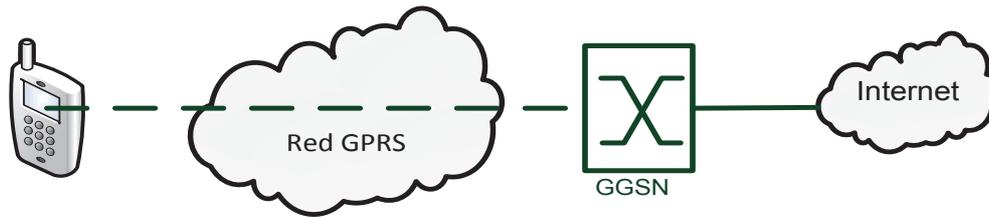
- Almacenar y mantener información en el registro de locación SLR de todos los subscriptores registrados en el área de enrutamiento relacionado a un SGSN particular.
- Funciones de gestión de sesión para la activación o desactivación del contexto PDP, para establecer una conexión a un determinado SGSN.
- Funciones de manejo de paquetes.
- Manejo del centro SMS conectado directamente al SGSN vía el interfaz Gd.
- Generación de archivos de facturación.
- Manejo y débito de las cuentas prepago para uso de GPRS.
- Gestión del rendimiento para medición del tráfico dentro de esta parte de la red GPRS.
- Mantenimiento y gestión de fallas para detectar problemas durante la transmisión.
- Manejo de protocolos e interfaces: Gb para conectarse a UCP; Gn para conectarse a GGSN, SGSN o DNS; Gr para conectarse a HLR; Gs para conectarse a MSC/VLR. Estas interfaces están implementados con módulos que funcionan con protocolos como: BSSGP, LLC, SNDTCP, GMM, SM y GTP.

#### *1.3.3.3.2. Nodo de Soporte de Acceso GPRS (Gateway GPRS Support Node GGSN) [7]*

El nodo de soporte de acceso GPRS (*Gateway GPRS Support Node GGSN*), permite la conexión de datos externos (intra-/Internet), con el núcleo de la red GPRS. Este nodo tiene como función establecer el Gateway entre el GPRS móvil y la red de paquete de datos externo (*Packet Data Network PDN*). La movilidad del MS es escondida por la GGSN ante el PDN, el cual ve al GGSN simplemente como un router adicional, con funcionalidades adicionales de log-in, como aquellos de un servidor de acceso de una red estándar.

Desde el punto de vista del MS, el procedimiento para establecer una conexión al internet, tal como se observa en la Figura 1.24 consiste en dos pasos:

- Primero, engancharse a la red GPRS lo cual involucra al SGSN.
- Segundo, establecimiento del contexto PDP vía SGSN al GGSN.



**Figura 1.24 Inicio de Sesión de un MS en un GGSN.**

*Fuente: [7]*

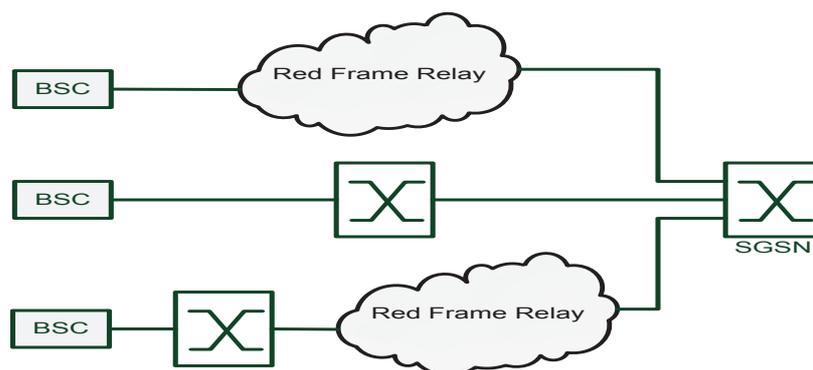
El GGSN es equivalente a un Gateway, que conecta a la MS con la red de datos. Las funciones del GGSN son las siguientes:

- Lectura de la cabecera de paquetes IP generados por la MS, para el envío de paquetes dentro de la red GPRS, basado en la dirección de destino obtenida.
- Funciones de manejo de paquetes, lo cual implican el envío de datos de entrada del usuario desde un SGSN al interfaz Gi y viceversa. También se da prioridad de conmutación de paquetes en el caso de calidad de servicio.
- Funciones de gestión de sesiones, para la activación o desactivación del contexto PDP para el establecimiento o liberación de una conexión con un determinado SGSN.
- Administración de usuarios, mediante la asignación de direcciones IP y DNS para los MS registrados.
- Generación de archivos de facturación.
- Gestión del rendimiento para medición del tráfico dentro de esta parte de la red GPRS.
- Mantenimiento y gestión de fallas para determinación de problemas durante la transmisión de datos.
- Manejo de interfaces como: Gn para conectarse con el SGSN, Gi para conectarse al PDN. Estos interfaces son implementados en módulos que utilizan protocolos como: GTP e IP

#### *1.3.3.3.3. Red de Acceso PCU –SGSN (Interfaz Gb) [7]*

Existen tres posibilidades de conexión entre el PCU y el SGSN como se puede observar en la Figura 1.25. A continuación se describen cada una de estas posibilidades:

- La solución más flexible es la utilización de una red Frame Relay (FR) pura, la cual está conformada por switches individuales interconectados entre sí. Cada switch es capaz de multiplexar canales virtuales de PCUs individuales en un solo recurso FR, el cual está conformado por un número de ranuras de tiempo (*time slots*) combinadas en un denominado grupo de canales, y a su vez, todas las conexiones comparten el mismo ancho de banda físico. Esto implica que una sola conexión puede hacer uso del ancho de banda completo, si las otras conexiones no transmiten ningún otro dato. Por tanto, un uso eficiente del ancho de banda es posible.
- Una segunda posibilidad para la interconexión entre el PCU y el SGSN, es la utilización de líneas arrendadas, que están conformadas por una sola o varias ranuras de tiempo (*time slot*), las cuales son multiplexadas individualmente en la capa 1 a través de la red. Por esta razón, no es posible multiplexar varias conexiones FR en un solo grupo de canales. El ancho de banda entre el PCU y el SGSN es fijo. De esta forma, una línea arrendada conformada por un grupo de canales, contiene solo una conexión desde un PCU hacia el SGSN.
- La tercera posibilidad es una mezcla entre la utilización de una red FR pura y la utilización de líneas arrendadas, con el fin de optimizar el ancho de banda y establecer una configuración flexible de las conexiones dentro de la red Frame Relay.



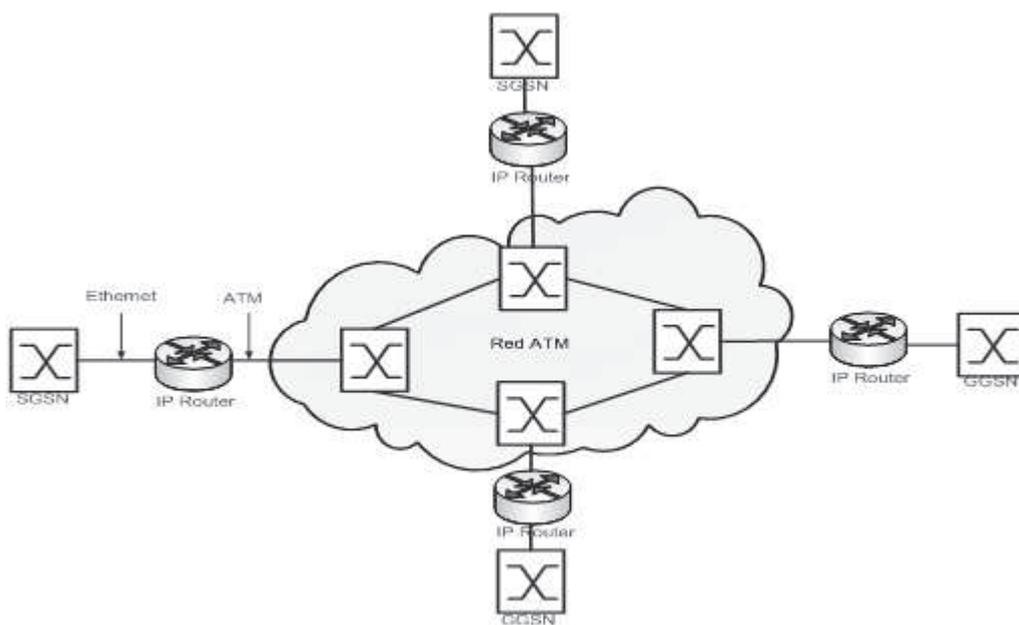
**Figura 1.25** Tres Posibilidades de Interconexión entre la PCU y la SGSN

Fuente: [7]

#### 1.3.3.3.4. Núcleo de Red SGSN, GGSN (Interfaz Gn) [7]

La interconexión entre la SGSN y GGSN dentro de la red, puede ser realizada mediante la utilización de una red ATM, como se muestra en la Figura 1.26. Se

observa que se utiliza routers basados en el protocolo IP. Los paquetes IP desde el SGSN/GGSN son transmitidos en tramas Ethernet al correspondiente router dentro de la red Gn. Normalmente se encuentra el SGSN/GGSN y el router en la misma locación, ambos presentan un interfaz Ethernet. El router accede a la red ATM para transmitir paquetes IP a otro SGSN/GGSN, por lo que el router debe contener interfaces basados en E1, E3, STM-1, etc, como protocolos de capa 1. Los routers son interconectados a través de la red ATM, y realizan funciones de conmutación usando conexiones permanentes.



**Figura 1.26 Estructura Típica de una Red Gn**

*Fuente: [7]*

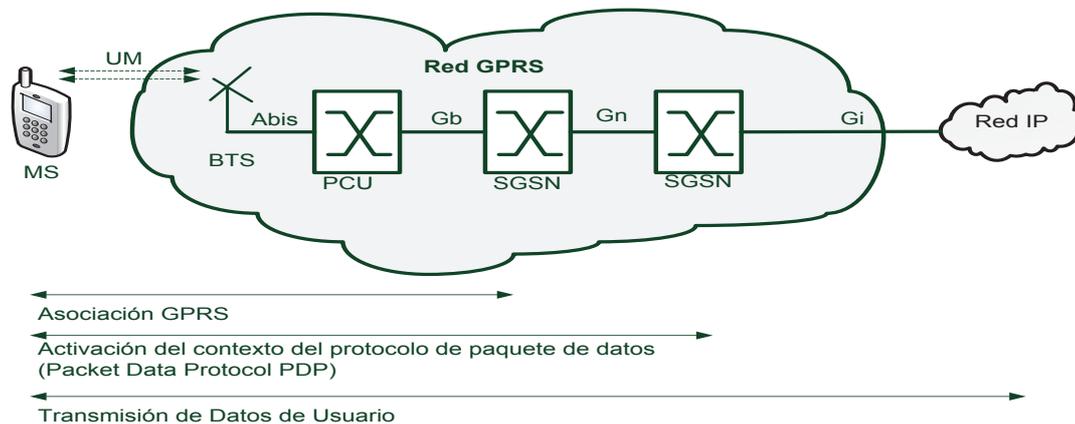
#### 1.3.4. PROCEDIMIENTO DE CONEXIÓN GPRS [8]

Antes de que la información pueda ser transmitida entre la estación móvil MS y la red de datos externa, se establecen procedimientos previos para habilitar la transferencia de paquetes IP a través de la red GPRS. Se mencionan entonces, tres pasos importantes que se muestran en la Figura 1.27.

1. El MS debe asociarse dentro de la red GPRS. El procedimiento denominado asociación GPRS, establece un proceso lógico entre la MS y el SGSN, el cual prescinde de la posición del MS. El almacenamiento y actualización de la

posición del MS es particularmente importante para la transmisión y localización del MS dentro de la red GPRS.

2. Determinación de la mejor ruta para la transmisión de paquetes IP dentro de la red GPRS. Una conexión entre el MS y el GGSN debe realizarse previamente, de tal forma que cada nodo dentro de la red GPRS conozca donde debe enviar el paquete IP de la MS. A este procedimiento se lo denomina activación del contexto del protocolo de paquetes de datos (Packet Data Protocol PDP).
3. Una vez activado el contexto PDP y designado la mejor ruta dentro de la red GPRS entre el MS y la red de datos externa, se envía los paquetes IP a través de la red GPRS hacia la dirección de destino. Este procedimiento contempla la transmisión de datos del usuario.



**Figura 1.27 Procedimientos GPRS.**

Fuente: [8]

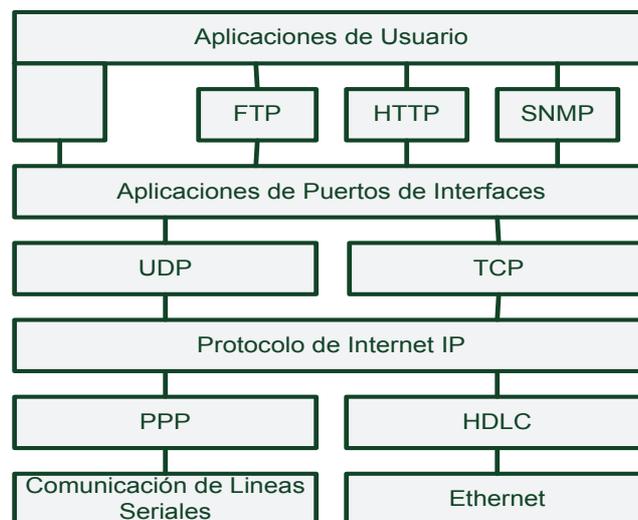
### 1.3.5. PROTOCOLO DE CONTROL DE TRANSMISIÓN (*TRANSMISSION CONTROL PROTOCOL TCP*) [9]

TCP es un servicio de comunicación orientada a la conexión que es responsable del establecimiento y liberación de un circuito virtual. TCP proporciona una comunicación segura entre pares de procesos (usuarios TCP), dentro de una red GPRS. Los servicios que TCP ofrece se definen en términos de primitivas y parámetros.

#### 1.3.5.1. Protocolo de Internet IP [10]

El protocolo de internet (*Internet Protocol IP*), es un protocolo no orientado a la conexión y no confiable, el cual no hace uso de acuses de recibo (ACK), ni

procedimientos de corrección de errores. Como resultado, se obtiene altas tasas de velocidades de transmisión. Sin embargo, la corrección de errores y la confiabilidad debe ser implementada en protocolos de capas superiores como es el caso de TCP. La tarea más importante de IP es el direccionamiento de equipos terminales mediante la utilización de direcciones IP y la fragmentación y ensamblaje de paquetes. La utilización de una arquitectura lógica mediante direcciones IP, permite procedimientos flexibles de enrutamiento. La ruta entre dos terminales de comunicación no es reservada previamente, en su lugar, se instaure un proceso de búsqueda de la mejor ruta en cada uno de los nodos, los cuales se encuentran en el trayecto entre los dos puntos de comunicación finales. Como se observa en la Figura 1.28, IP permite encapsular protocolos de capa de transporte como UDP y TCP, así como protocolos de capa de aplicación como FTP, HTTP, SNMP, y muchos más que funcionan bajo los protocolos de capa 4 mencionados previamente.



**Figura 1.28** Conjunto de Protocolos compatibles con IP.

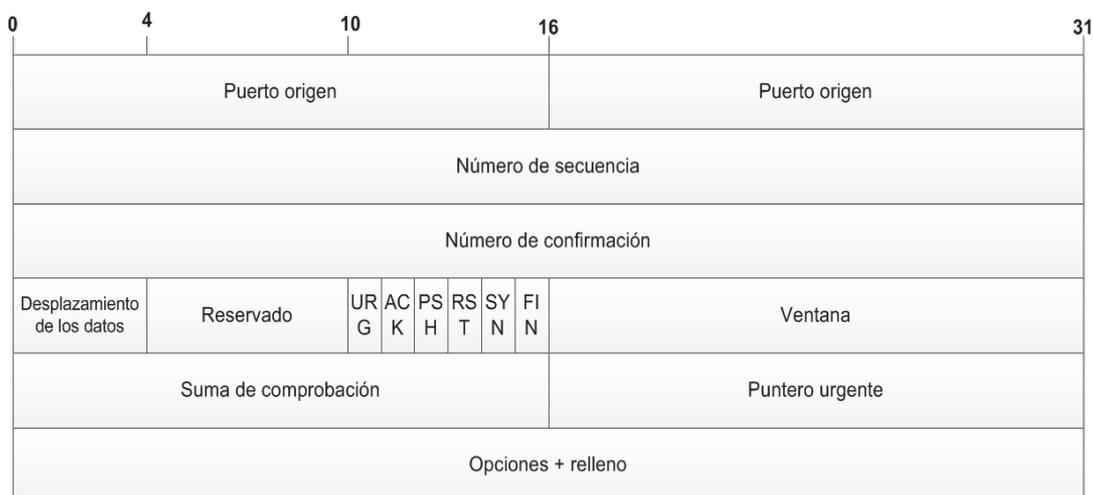
*Fuente: [10]*

### 1.3.5.2. Formato de la Cabecera TCP [9]

La unidad de datos de protocolo de TCP, se denomina segmento TCP mostrada en la Figura 1.29. La cabecera presenta un longitud de 20 octetos o 20-bytes. Los campos son los siguientes:

- Puerto origen (16-bits): usuario TCP origen.

- Puerto destino (16-bits): usuario TCP destino.
- Número de secuencia (32-bits): número de secuencia del primer octeto en este segmento.
- Número de confirmación (32-bits): Contiene el número de secuencia del siguiente octeto que la entidad TCP espera recibir.
- Longitud de cabecera (4-bits): número de palabras de 32-bits en la cabecera.
- Reservados (6-bits): bits reservados para uso futuro.
- Indicadores (6-bits):
  - URG: el campo puntero urgente es válido.
  - ACK: el campo de confirmación es válido.
  - PSH: función de carga.
  - RST: puesta a cero de la conexión.
  - SYN: sincronizar el número de secuencia.
  - FIN: el emisor no tiene más datos
- Ventana (16-bits): utilizado para control de flujo.
- Suma de verificación (16-bits): el complemento a uno de la suma módulo  $2^{16} - 1$  de todas las palabras de 16-bits en el segmento más una pseudo-cabecera, descrita más abajo.
- Puntero urgente (16-bits): señala el octeto que sigue a los datos urgentes. Esto permite al receptor conocer cuántos datos urgentes llegan.
- Opciones (Variable): solamente se define una opción, que especifica el tamaño del segmento que será aceptado.



**Figura 1.29 Cabecera TCP.**

Fuente: [9]

El campo de suma de verificación se aplica a todo el segmento entero más una pseudo-cabecera incorporada en el momento del cálculo (tanto en transmisión como en recepción). La pseudo-cabecera incluye los siguientes campos de la cabecera IP: dirección internet origen y destino, el protocolo y un campo longitud del segmento. Con la inclusión de la pseudo-cabecera, TCP se protege a sí mismo de una transmisión errónea de IP. Esto es, si IP lleva un segmento a un computador erróneo, aunque el segmento esté libre de errores, la entidad TCP receptora detectará el error de transmisión.

## CAPÍTULO 2

### REQUERIMIENTOS DEL PROTOTIPO

#### 2.1. DESCRIPCIÓN GENERAL DEL INAMHI

El Instituto Nacional de Meteorología e Hidrología (INAMHI), se encarga de coordinar y normalizar la política hidrometeorológica a nivel nacional. Es decir, tiene la facultad de implementar, operar y dar mantenimiento a Estaciones Hidrometeorológicas, con el objetivo de generar información básica y suministrar servicios y productos hidrometeorológicos necesarios para el desarrollo del país. El INAMHI, se encuentra adscrito a la Secretaría Nacional de Riesgo, y fue creado el 4 de agosto de 1961, como una institución de investigación hidrometeorológica [11].

##### 2.1.1. FUNCIÓN DEL INAMHI.

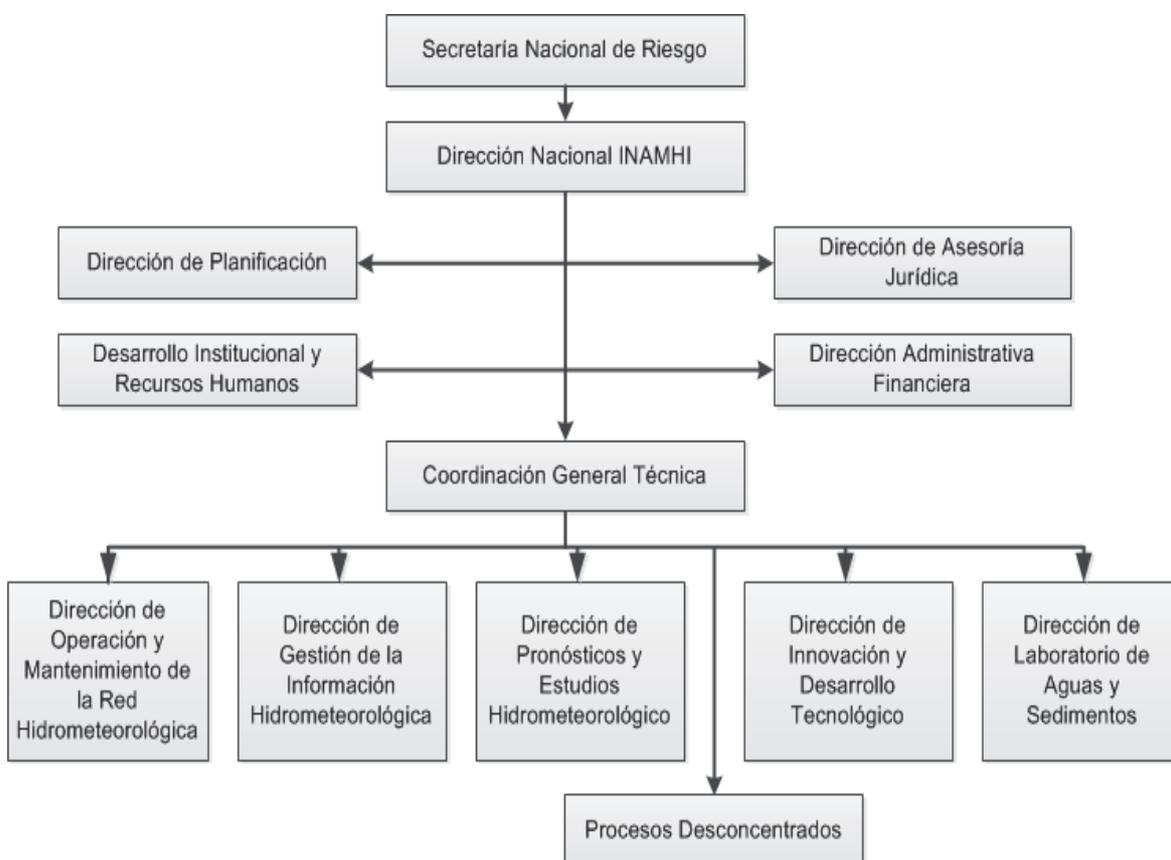
Según la página web oficial del INAMHI [www.inamhi.gob.ec](http://www.inamhi.gob.ec) (2012), la misión del INAMHI es:

*“El INAMHI es la entidad técnico - científica responsable en el Ecuador de la generación y difusión de la información hidrometeorológica que sirva de sustento para la formulación y evaluación de los planes de desarrollo nacionales y locales y la realización de investigación propia o por parte de otros actores, aplicada a la vida cotidiana de los habitantes y los sectores estratégicos de la economía; apoyado en personal especializado y en una adecuada utilización de las nuevas tecnologías de la automatización, información y comunicación”*

De acuerdo a la página web oficial del INAMHI [www.inamhi.gob.ec](http://www.inamhi.gob.ec) (2012), las funciones del INAMHI se definen en el siguiente párrafo:

*“El INAMHI mantiene la representación del Ecuador en la Organización Meteorológica Mundial – OMM- de ahí que sus actividades se orientan a: mejorar la calidad y cobertura de la información que se genera a través de la red de estaciones hidrometeorológicas; mantener actualizada la base de datos; facilitar el acceso de los usuarios a la información técnica y a los servicios que la institución oferta a la comunidad; posibilitar y mejorar el suministro de avisos de alerta de fenómenos hidrometeorológicos adversos asegurar que lleguen al público; fortalecer la capacidad de prestación de servicios y proveer productos cada vez más útiles sobre el tiempo, el clima y el medio ambiente; informar al público acerca de los beneficios socioeconómicos derivados de la mejor comprensión del clima y de la distribución de los recursos hídricos y trabajar con mayor eficacia y coordinación con instituciones públicas y privadas del país, y del exterior”*

### 2.1.2. ORGANIGRAMA DEL INAMHI

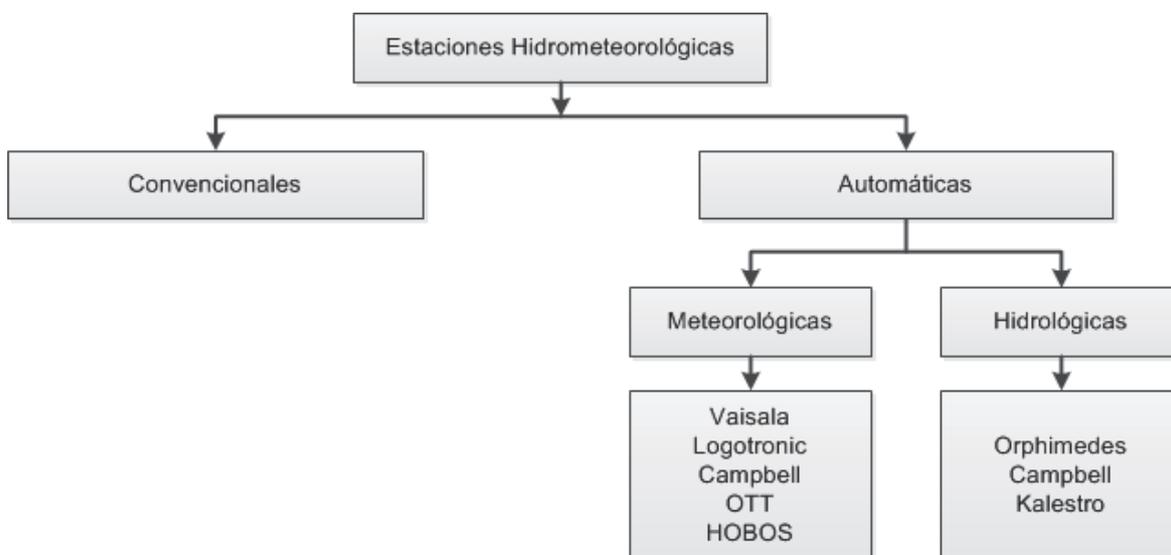


**Figura 2.1 Organigrama del Instituto Nacional de Meteorología e Hidrología del Ecuador INAMHI**  
Fuente: [12]

El organigrama del INAMHI se muestra en la Figura 2.1. Se debe mencionar que la instalación y mantenimiento de las estaciones convencionales está a cargo de la dirección de operación y mantenimiento de la red hidrometeorológica y la instalación y mantenimiento de las estaciones automáticas, está a cargo de la dirección de innovación y desarrollo tecnológico.

### 2.1.3. CLASIFICACIÓN DE LAS ESTACIONES HIDROMETEOROLÓGICAS UTILIZADAS POR EL INAMHI

Las estaciones hidrometeorológicas que el INAMHI ha instalado, pueden ser clasificadas como: convencionales y automáticas, tal como se muestra en el diagrama de la Figura 2.2.



**Figura 2.2 Diagrama de Clasificación de las Estaciones Hidrometeorológicas**  
Fuente: Elaboración propia

#### 2.1.3.1. Estaciones Climatológicas Convencionales

Una estación climatológica convencional es aquella en la cual se realizan observaciones de precipitación, temperatura del aire, humedad, viento, radiación solar, evaporación, cantidad de nubes y fenómenos especiales. Las variables mencionadas, pueden ser obtenidas mediante instrumentos registradores o instrumentos de lectura y sensores, como se detalla en la Tabla 2.1.

La confiabilidad de la medida de los elementos meteorológicos, depende de la precisa instalación de los instrumentos, de acuerdo a la distribución de los

mismos en el sitio donde se encuentre ubicada la estación climatológica, tal como se muestra en la Figura 2.3. Además, para obtener información comparable, es requisito indispensable que la instalación de los instrumentos sea semejante en todas las estaciones, y que las observaciones meteorológicas sean realizadas con puntualidad. La estación climatológica debe estar situada en lugar y condiciones que garanticen la operación regular de la estación durante un largo período de tiempo [13].

**Tabla 2.1 Instrumentos de Lectura y Registradores de Variables Hidrometeorológicas**

Parámetro	Instrumento Lectura Directa y Sensores	Instrumento Registrador
Temperatura del aire	Termómetro	Termógrafo
Temperatura del agua y del suelo	Termómetro	Batitermógrafo y geotermógrafo
Humedad atmosférica	Higrómetro	Hidrógrafo
Precipitación	Pluviómetro	Pluviógrafo
Evaporación	Tanque de evaporación	Evaporígrafo
Velocidad y dirección del viento	Anemómetro y veleta	Anemógrafo
Insolación		Heliógrafo
Radiación	Piranómetro	Actinógrafo

*Fuente: [13]*



**Figura 2.3 Estación Climatológica Convencional**

*Fuente: [12]*

Dentro de la caseta, se pueden ubicar instrumentos medidores de temperatura y humedad como: sigrómetro, termómetro de máxima y termómetro de mínima, así como los correspondientes registradores: termógrafo e higrógrafo. La Figura 2.4, muestra la caseta meteorológica y los instrumentos que en ella se encuentran.



**Figura 2.4 Caseta Meteorológica**

*Fuente: [12]*

### **2.1.3.2. Estaciones Automáticas**

Una estación automática no es más que un sistema conformado por un conjunto de elementos, que permiten el manejo y procesamiento de la información hidrometeorológica de forma autónoma. Además, una estación de este tipo es de bajo consumo de energía y utiliza baterías y paneles solares para su respectiva energización.

Una estación meteorológica automática mostrada en la Figura 2.5, está conformada por los siguientes elementos:

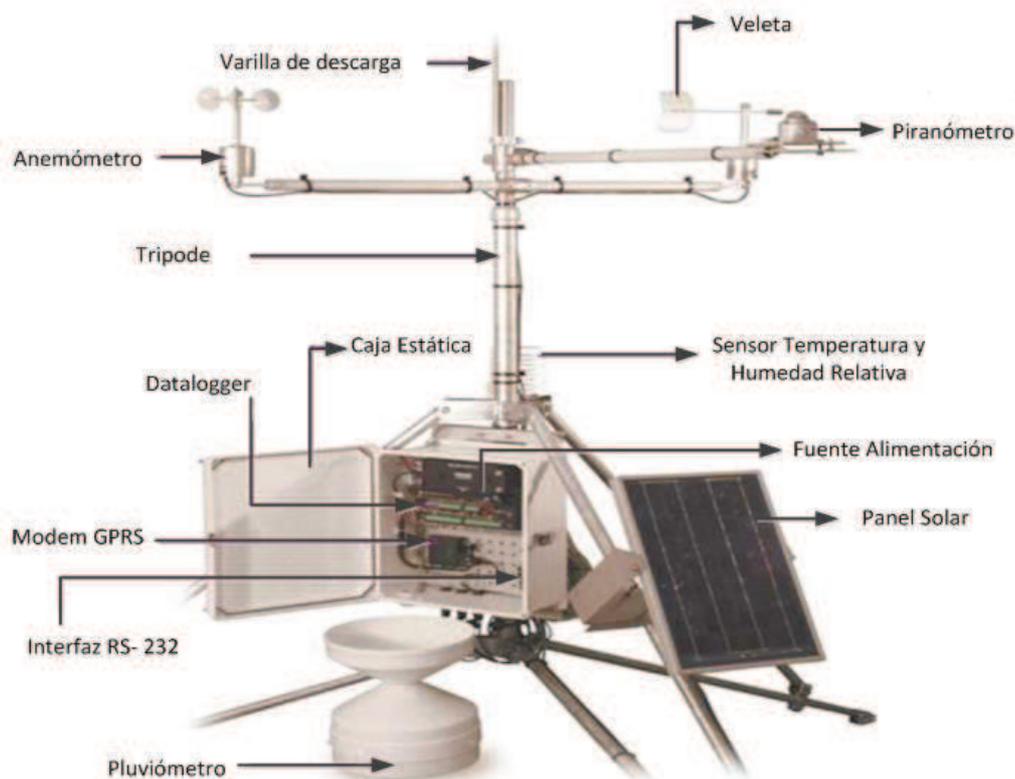
- Adquisidor de datos (*Data Logger*).
- Transmisor de datos.
- Sensores.
- Sistema de energización.
- Infraestructuras de soporte.

Las estaciones automáticas pueden ser clasificadas como: meteorológicas e hidrológicas.

#### *2.1.3.2.1. Estaciones Automáticas Meteorológicas*

Una estación automática meteorológica, se encarga de recopilar y procesar datos de: temperatura relativa, humedad relativa, precipitación, radiación solar,

velocidad y dirección del viento, para lo cual hacen uso de diferentes tipos de sensores, de acuerdo al tipo de parámetro de medición tal como se mostró en la Tabla 2.1. El INAMHI, hace uso de estaciones de los fabricantes: Vaisala, Logotronic, Campbell, OTT y HOBOS.



**Figura 2.5 Partes de una Estación Meteorológica Campbell**  
Fuente: [14]

#### 2.1.3.2.2. Estaciones Automáticas Hidrológicas

Una estación automática hidrológica, se encarga de la recopilación y procesamiento de datos de: precipitación, caudal líquido, caudal sólido y calidad del agua. El INAMHI, hace uso de estaciones hidrológicas de los siguientes tipos: Orphimedes, Campbell y Kalestro.

#### 2.1.4. RED DE ESTACIONES AUTOMÁTICAS METEOROLÓGICAS INSTALADAS [15]

El INAMHI presenta un total de 100 estaciones automáticas de diferentes marcas instaladas, de las cuales, 67 son estaciones meteorológicas, y 33 hidrológicas..

El mantenimiento, operación e instalación de cada una de ellas es realizada por la unidad de Innovación y Desarrollo Tecnológico (IDT) en Quito y al Proceso Desconcentrado en Guayaquil. El IDT tiene la responsabilidad sobre 62 estaciones distribuidas en la Sierra y Oriente ecuatoriano, mientras que el proceso desconcentrado en Guayaquil tiene la responsabilidad sobre 29 estaciones distribuidas en la Costa ecuatoriana. Además, la EPMAPS realiza la operación y mantenimiento de 7 estaciones pluviométricas en el Antisana.

## **2.2. NECESIDAD Y SOLUCIÓN REFERENTE A LA IMPLEMENTACIÓN DE UNA ESTACIÓN AUTOMÁTICA INALÁMBRICA POR PARTE DEL INAMHI**

De acuerdo a la descripción realizada en la secciones 2.1.3 y 2.1.4, correspondiente a la clasificación de la estaciones hidrometeorológicas que el INAMHI utiliza, se observa que este instituto no cuenta con estaciones automáticas inalámbricas, refiriéndose a las mismas como aquellas que permiten adquirir y procesar las distintas variables meteorológicas de forma inalámbrica, facilitando así la movilidad y ubicación de los distintos sensores dentro de un determinado rango de cobertura en campo o en el punto de medición. En este sentido, se ve la necesidad de implementar una solución alternativa y propia, para la adquisición, procesamiento y transmisión inalámbrica de la información meteorológica por parte del INAMHI, y que además, sea de bajo costo.

Por lo tanto, la solución que se plantea es el diseño e implementación de una red inalámbrica de área personal (Wireless Personal Area Network WPAN), cuyos requerimientos sean de bajo consumo de batería, bajas velocidades de transmisión, bajo costo y larga vida de funcionamiento, que permita la comunicación inalámbrica, ubicación y movilidad de los distintos sensores meteorológicos utilizados en los puntos de medición, para la correspondiente adquisición de datos. Se presenta entonces, a los estándares IEEE 802.15.4 y ZigBee, como aquellos que permiten cumplir con los requerimientos de una red inalámbrica de área personal previamente planteados. Además, se debe mencionar que la transmisión por parte de la estación remota hacia el servidor para tratamiento de la información meteorológica, se realizará utilizando la misma

solución funcional y que el INAMHI ha implementado mediante comunicación GPRS.

Adicionalmente, se debe mencionar que las únicas variables meteorológicas a ser tomadas en cuenta para su correspondiente procesamiento y medición dentro del diseño de este prototipo, son las de temperatura, humedad y precipitación.

### **2.3. DETERMINACIÓN DE LAS ESPECIFICACIONES DEL PROTOTIPO**

En esta sección se describe la secuencia de pasos que se siguen para determinar las especificaciones del prototipo tomando en cuenta la solución presentada en la sección 2.2. En estos términos, se considera entonces los siguientes parámetros a ser determinados:

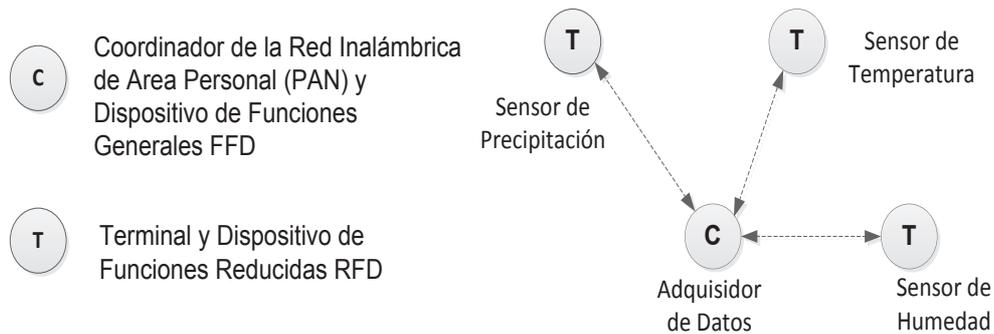
- 1) Topología de la red inalámbrica ZigBee e IEEE 802.15.4 presentada como solución.
- 2) Módulos de prototipo y sus correspondientes funciones.
- 3) Variables de entrada y salida de cada uno de los módulos, de acuerdo a las funciones instauradas previamente.
- 4) Especificaciones del prototipo en base a las variables mencionadas en el punto anterior.

#### **2.3.1. TOPOLOGÍA DE RED DEL PROTOTIPO**

Tomando en cuenta que la solución propuesta para el diseño e implementación de una red inalámbrica de área personal es la utilización de los estándares IEEE 802.15.4 y ZigBee, se debe especificar la topología de red que se debe implementar en el campo o punto de medición, en base a las funciones que se deseen cumplir.

En este sentido, se indica que una estación automática consta de un adquisidor de datos encargado de recolectar la correspondiente información, y de una serie de sensores encargados de captar y enviar las mediciones de las variables meteorológicas al adquisidor de datos. Se puede deducir entonces, que la

necesidad de una topología tipo estrella debe ser implementada, en la cual debe existir un dispositivo coordinador cuya función debe ser parte del módulo adquisidor de datos y de igual manera, los dispositivos finales o terminales, deben ser parte de los distintos módulos que contengan a los sensores, tal como se observa en la Figura 2.6.



**Figura 2.6 Topología tipo Estrella de la Red Inalámbrica del Prototipo**

*Fuente: Elaboración propia*

### 2.3.2. FUNCIONES Y DEFINICIÓN DE LOS MÓDULOS DEL PROTOTIPO

En base a la definición de una topología tipo estrella para una red inalámbrica ZigBee, se definen a los siguientes módulos que permitirán el correspondiente procesamiento, adquisición y presentación de la información de temperatura, humedad y precipitación:

- a) Módulo de Temperatura y Humedad Relativa.
- b) Módulo de Precipitación.
- c) Módulo de Adquisición de Datos Meteorológicos.
- d) Módulo de Transmisión y Procesamiento de Datos GPRS.

#### 2.3.2.1. Funciones del Módulo de Temperatura y Humedad Relativa

Este módulo presenta las siguientes funciones:

- Acondicionar las señales analógicas del sensor de temperatura y humedad relativa.
- Detectar el nivel de voltaje necesario para que el circuito funcione.
- Energización y regulación de voltaje a valores funcionales del circuito.

- Transmisión inalámbrica de datos de temperatura y humedad relativa al dispositivo ZigBee Coordinador, y que a su vez, permita la recepción de datos procesados del dispositivo ZigBee Coordinador.
- Presentación de los datos de nivel de voltaje, temperatura y humedad relativa en un display.

#### **2.3.2.2. Funciones del Módulo de Precipitación**

En este módulo se define las siguientes funciones:

- Acondicionar las señales del sensor de precipitación.
- Detectar el nivel de voltaje necesario para que el circuito funcione.
- Energización y regulación de voltaje a valores funcionales del circuito.
- Transmisión inalámbrica de datos de precipitación al dispositivo ZigBee Coordinador, y que a su vez, permita la recepción de datos procesados del dispositivo ZigBee Coordinador.
- Presentación de los datos de nivel de voltaje y precipitación en un display.

#### **2.3.2.3. Funciones del Módulo de Adquisición de Datos**

Este módulo define las siguientes funciones:

- Procesar, y presentar datos meteorológicos de precipitación, temperatura y humedad relativa adquiridos por el dispositivo coordinador ZigBee.
- Recopilar y almacenar la información de nivel de voltaje de funcionamiento de los circuitos e información meteorológica de precipitación, temperatura y humedad relativa.
- Administrar, gestionar y controlar la red inalámbrica ZigBee.

#### **2.3.2.4. Funciones del Módulo de Transmisión y Procesamiento de Datos GPRS**

En este módulo se define las siguientes funciones:

- Transmisión de datos de nivel de voltaje necesario para funcionamiento de los módulos e información meteorológicos de precipitación, temperatura y

humedad relativa, procesados por el módulo de adquisición de datos, al servidor localizado en el INAMHI Quito, vía GPRS.

- Establecimiento de una comunicación punto a punto entre la estación remota y un servidor localizado en el INAMHI, haciendo uso de la red de datos GPRS de movistar o claro.
- Almacenamiento y tratamiento de los datos enviados vía GPRS por medio de un servidor.

### **2.3.3. VARIABLES DE ENTRADA Y SALIDA DE CADA UNO DE LOS MÓDULOS DEL PROTOTIPO**

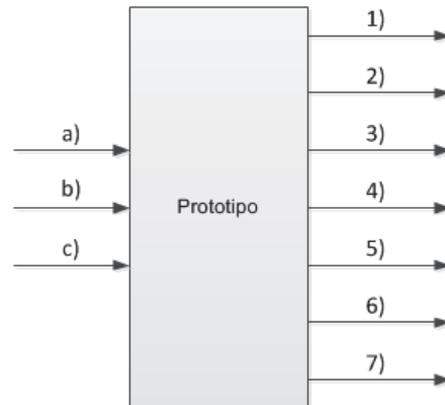
Tomando en cuenta los módulos y sus correspondientes funciones definidos en la sección 2.3.2, se procede a determinar las variables de entrada y salida. Inicialmente se considera a todo el sistema y prototipo en general tal como se muestra en la Figura 2.7, para luego especificar las correspondientes variables de entrada y salida de forma específica en cada uno de los módulos.

De esta manera, las variables de entrada del prototipo son:

- a) Señales del sensor de temperatura.
- b) Señales del sensor de humedad.
- c) Señales del sensor de precipitación.

Por otro lado, las variables de salida del prototipo son:

- 1) Valores digitales de temperatura.
- 2) Valores digitales de humedad.
- 3) Valores digitales de precipitación.
- 4) Valores digitales del nivel de voltaje necesario para funcionamiento de los distintos módulos.
- 5) Valores digitales de los niveles de señal de los enlaces ZigBee inalámbricos.
- 6) Gráficas de temperatura, humedad y precipitación vs tiempo.
- 7) Tablas de recopilación de información por fecha y hora de temperatura, humedad y precipitación.

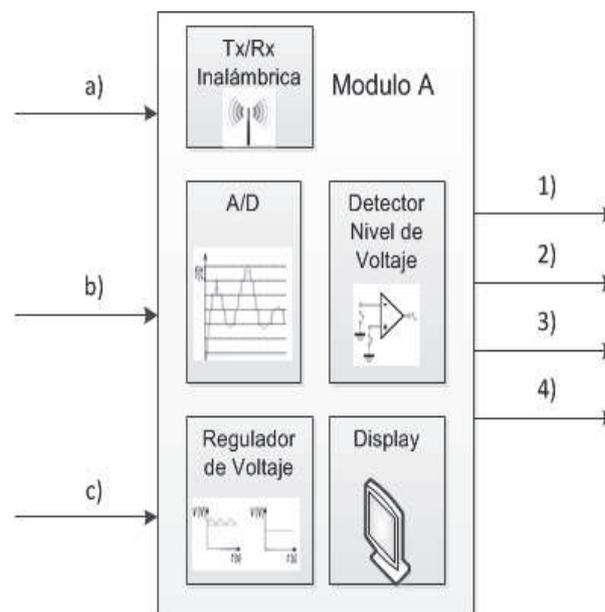


**Figura 2.7 Variables de Entrada y Salida del Prototipo**

*Fuente: Elaboración propia*

### 2.3.3.1. Variables del Módulo de Temperatura y Humedad Relativa

En base a las funciones definidas en la sección 2.3.2.1 para este módulo, se procede a enumerar las correspondientes variables de entrada y salida del mismo, tal como se observa en la Figura 2.8.



**Figura 2.8 Funciones y Variables de Entrada y Salida del Módulo de Temperatura y Humedad Relativa**

*Fuente: Basado en las funciones de la sección 2.3.2.1.*

Variables de entrada:

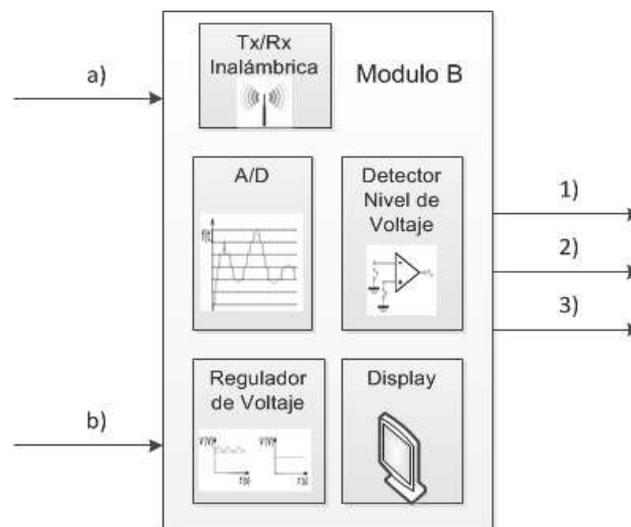
- a) Sensor de temperatura.
- b) Sensor de humedad.
- c) Voltaje necesario para alimentar al circuito.

Variables de salida:

- 1) Valores de voltaje del sensor de temperatura.
- 2) Valores de voltaje del sensor de humedad.
- 3) Valores de voltaje de funcionamiento del circuito.
- 4) Valores de los parámetros del enlace ZigBee inalámbrico.

### 2.3.3.2. Variables del Módulo de Precipitación

En base a las funciones definidas en la sección 2.3.2.2 para este módulo, se procede a enumerar las correspondientes variables de entrada y salida del mismo, tal como se observa en la Figura 2.9.



**Figura 2.9 Funciones y Variables de Entrada y Salida del Módulo de Precipitación.**

*Fuente: Basado en las funciones de la sección 2.3.2.2.*

Variables de entrada:

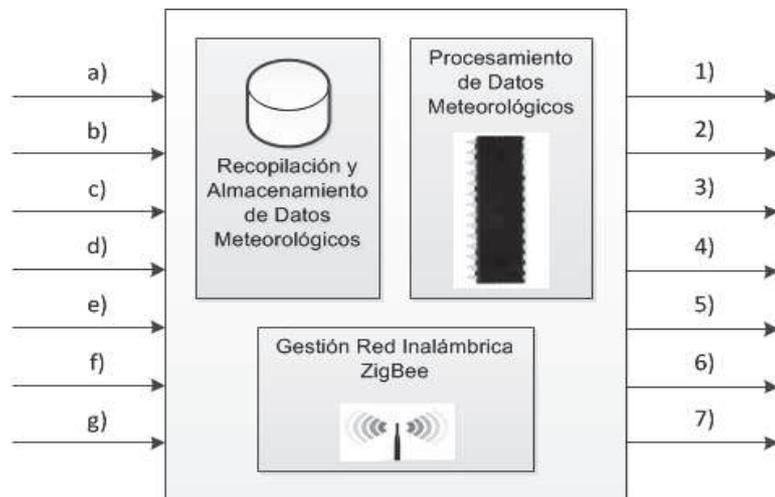
- a) Sensor de precipitación.
- b) Voltaje necesario para alimentar el circuito.

Variables de salida:

- 1) Valores de las basculaciones del sensor de precipitación.
- 2) Valores de voltaje de funcionamiento del circuito.
- 3) Valores de los parámetros del enlace ZigBee inalámbrico.

### 2.3.3.3. Variables del Módulo de Adquisición de Datos

De acuerdo a las funciones definidas en la sección 2.3.2.3 para este módulo, se procede a enumerar las correspondientes variables de entrada y salida del mismo, tal como se observa en la Figura 2.10.



**Figura 2.10 Funciones y Variables de Entrada y Salida del Módulo de Adquisición de Datos.**

*Fuente: Elaborado en base a las funciones de la sección 2.3.2.3.*

Variables de entrada:

- a) Valores de voltaje del sensor de temperatura.
- b) Valores de voltaje del sensor de humedad.
- c) Valores de las basculaciones del sensor de precipitación.
- d) Valores de voltaje de funcionamiento del circuito A.
- e) Valores de voltaje de funcionamiento del circuito B.
- f) Valores de los parámetros del enlace ZigBee inalámbrico del módulo A.
- g) Valores de los parámetros del enlace ZigBee inalámbrico del módulo B.

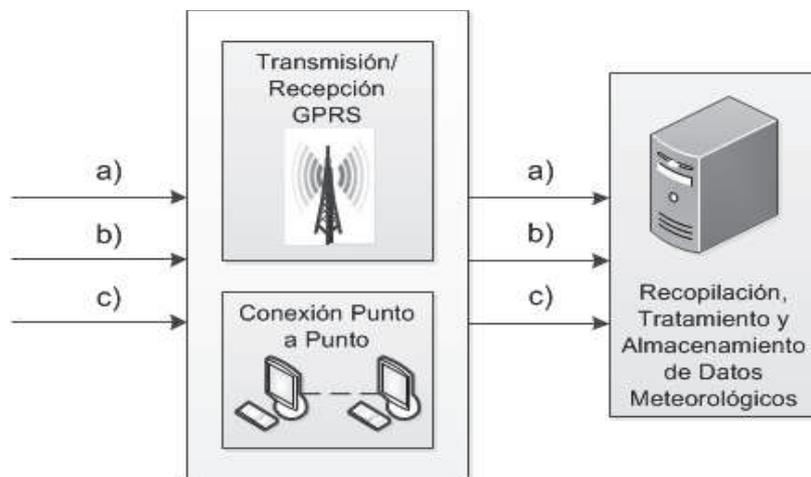
Variables de salida:

- 1) Valores de temperatura.
- 2) Valores de humedad.
- 3) Valores de precipitación.
- 4) Porcentaje de nivel de batería del módulo A.
- 5) Porcentaje del nivel de batería del módulo B.

- 6) Valores de nivel de señal de los enlaces ZigBee inalámbricos.
- 7) Estado y alarmas de las mediciones de temperatura, humedad y precipitación.

#### 2.3.3.4. Variables del Módulo de Transmisión y Procesamiento de Datos GPRS

Tomando en cuenta las funciones definidas en la sección 2.3.2.4 para este módulo, se procede a enumerar las correspondientes variables de entrada y salida, tal como se muestra en la Figura 2.11.



**Figura 2.11 Funciones y Variables de Entrada y Salida del Módulo de Transmisión y Procesamiento de Datos GPRS.**

*Fuente: Elaborado en base a las funciones de la sección 2.3.2.4.*

Para este caso las variables de entrada son las mismas que las variables de salida, ya que únicamente se está procediendo al correspondiente envío de información vía GPRS.

- a) Valores de temperatura.
- b) Valores de humedad.
- c) Valores de precipitación.

#### 2.3.4. ESPECIFICACIONES DEL PROTOTIPO

Tomando en cuenta las funciones y las correspondientes variables de entrada y salida descritas en las secciones 2.3.2 y 2.3.3, se procede a especificar los dispositivos y equipos necesarios, de acuerdo a la Tabla 2.2.

**Tabla 2.2 Especificaciones de Equipos y Dispositivos necesarios, de acuerdo a las Funciones de cada Módulo del Prototipo**

<b>Módulos de Prototipo</b>	<b>Especificaciones de cada Módulo del Prototipo</b>	<b>Equipos y Dispositivos Requeridos</b>
Módulo de temperatura y humedad relativa y módulo de precipitación	Energización y alimentación del circuito	Panel solar y batería DC
	Obtención de las señales de temperatura y humedad relativa	Sensor de temperatura y humedad relativa
	Obtención de las señales de precipitación	Sensor de precipitación
	Convertor análogo digital y detector de nivel de voltaje	Microprocesador
	Presentación de información digital	Display LCD
	Transmisión y recepción de información inalámbrica dentro de la red de área personal PAN	Dispositivo terminal ZigBee.
	Regulación de voltaje	Reguladores para el microcontrolador y el dispositivo ZigBee
Módulo de adquisición de datos	Recopilación y almacenamiento de datos meteorológicos	Computador con capacidad suficiente para almacenar datos tipo texto
	Procesamiento de datos meteorológicos	Software de procesamiento y tratamiento de las señales adquiridas, instalado en el computador
	Gestión de la red inalámbrica de área personal ZigBee	Dispositivo coordinador ZigBee
Módulo de transmisión y procesamiento de datos GPRS	Transmisión y recepción de datos GPRS	Modem GPRS
	Conexión punto a punto	Computador cliente y un computador servidor
	Recopilación, tratamiento y almacenamiento de datos meteorológicos	Servidor que utilice un software de procesamiento de datos meteorológicos

*Fuente: Elaboración propia.*

## **2.4. REQUERIMIENTOS DE HARDWARE Y SOFTWARE PARA IMPLEMENTACIÓN DEL PROTOTIPO**

En base a las especificaciones del prototipo descritas previamente, se definen características técnicas, que permitan seleccionar los dispositivos y equipos que cumplan con las funciones predefinidas para cada módulo del prototipo.

### **2.4.1. DISPOSITIVOS Y CARACTERÍSTICAS TÉCNICAS DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA**

De acuerdo a los dispositivos especificados en la Tabla 2.2 para este módulo, se indica que se utilizará un display LCD de 4x20 caracteres, reguladores de voltaje

tanto para el microprocesador y dispositivos ZigBee, y además se debe mencionar que los dispositivos de energización tales como panel solar y batería y de regulación de voltaje, se procederán a determinar posteriormente a su correspondiente dimensionamiento presentado en el capítulo 3 de diseño. Por otro lado, los dispositivos tales como sensor de temperatura y humedad relativa, microprocesador y dispositivos ZigBee, se presentan algunas alternativas, de las cuales se procede a seleccionar la mejor, en base a las características técnicas definidas.

#### 2.4.1.1. Sensor de Temperatura y Humedad Relativa

Las características técnicas y opciones que se presentan para este sensor se observan en la Tabla 2.3. Adicionalmente, se indica que estos sensores son provistos por el INAMHI.

**Tabla 2.3 Características Técnicas de los Sensores de Temperatura y Humedad Relativa**

Características Técnicas	HMP45A	HMP45D
Tipo de sensor	Pt 1000	Pt 100
Voltaje de alimentación	7.....35VDC	7.....35VDC
Consumo de energía	<4mA	<4mA
Rango de temperatura de operación	-40.....+60°C	-40.....+60°C
Rango de humedad de operación	0.8.....100%RH	0.8.....100%RH
Escala de salida	0.....1V = -40.....+60°C 0.....1V = 0.....100%RH	En función de la conexión resistiva de cuatro terminales
Tolerancia de error	±0.2°C, ±0.2%RH	Depende de la tolerancia de la resistencia

Fuente: [16].

Se selecciona el sensor de temperatura y humedad relativa HMP45A mostrado en la Figura 2.12 del fabricante VAISALA, debido a su facilidad de medición que presenta en la correspondiente escala de salida.



**Figura 2.12 Sensor de Temperatura y Humedad Relativa HMP45A/D de VAISALA**

Fuente: [16]

### 2.4.1.2. Microcontrolador

Las características técnicas mínimas y opciones que se presentan para la selección del microcontrolador se presentan en Tabla 2.4.

**Tabla 2.4 Características Técnicas Mínimas de Microcontroladores**

<b>Características Técnicas</b>	<b>ATmega164P</b>	<b>ATmega8</b>
Memoria Flash	16K Bytes	8K Bytes
Canales ADC	8 de 10-bits ADC	6 de 10-bits
Temporizadores	1 de 8-bits y 2 de 16-bits	2 de 8-bits y 1 de 16-bits
Interfaz serial	USART, SPI, JTAG	USART, SPI
Consumo de energía	0.4mA	3.6mA en modo activo

Fuente: [17]

Se escoge el microcontrolador ATmega164P mostrado en la Figura 2.13, debido a que presenta características suficientes y necesarias para el procesamiento y conversión analógica-digital de las señales de temperatura y humedad, y para el correspondiente detección del nivel de voltaje para funcionamiento del circuito.



**Figura 2.13 Microcontrolador ATmega164P del Fabricante Atmel**

Fuente: [17]

### 2.4.1.3. Dispositivo Terminal ZigBee

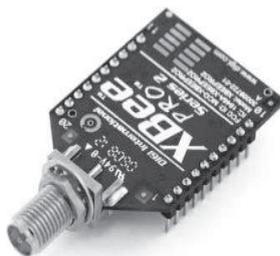
Las características técnicas mínimas y opciones que se presentan para la selección de los dispositivos ZigBee se presentan en Tabla 2.5.

**Tabla 2.5 Características Técnicas Mínimas de los Dispositivos Terminales ZigBee.**

<b>Características Técnicas</b>	<b>XBee</b>	<b>XBee-PRO</b>
Rango de cobertura indoor	Hasta los 30m	Hasta los 90m
Rango de cobertura outdoor	Hasta los 90m	Hatas 1600m
Potencia de transmisión	1mW	63mW
Velocidad de transmisión	250 Kbps	250 Kbps
Sensibilidad de recepción	-92dBm	-100dBm
Número de canales	16	12

Fuente: [18].

Se selecciona al dispositivo XBee-PRO mostrado en la Figura 2.14, como módulo de transmisión/recepción ZigBee, ya que presenta mejores características de cobertura y potencia de transmisión.



**Figura 2.14** Módulo de Tx/Rx XBEE-PRO

Fuente: [18]

## 2.4.2. DISPOSITIVOS Y CARACTERÍSTICAS TÉCNICAS DEL MÓDULO DE PRECIPITACIÓN

Este módulo, al igual que el módulo de temperatura y humedad relativa, presenta los mismos dispositivos y equipos, con excepción del sensor de precipitación. Por lo tanto, en esta sección únicamente se presenta características de este sensor.

### 2.4.2.1. Sensor de Precipitación

Las características técnicas y opciones que se presentan para la selección del sensor de precipitación se presentan en Tabla 2.6. Además, se indica que los sensores de precipitación son provistos por el INAMHI.

**Tabla 2.6** Características Técnicas del Sensor de Precipitación

Características Técnicas	Texas Electronics TR-525	Young 52202
Resolución	0.1 mm	0.1 mm
Precisión	1.0% a 10 mm por hora	2% a 25 mm por hora
Capacidad máxima que soporta el switch	30VDC @ 2A, 115VAC @ 1ª	24VAC/DC @ 500mA

Fuente: [19]

Se selecciona el sensor TR-525 de Texas Electronics mostrado en la Figura 2.15, debido a que presenta mayor capacidad de soporte de voltaje y mejor precisión.



**Figura 2.15** Sensores de Precipitación de la Serie 525

Fuente: [19]

### 2.4.3. DISPOSITIVOS Y CARACTERÍSTICAS TÉCNICAS DEL MÓDULO DE ADQUISICIÓN DE DATOS

En este módulo es necesario la utilización de un computador que permita el procesamiento, almacenamiento y presentación de los datos meteorológicos de precipitación, temperatura y humedad relativa recopilados por el dispositivo coordinador ZigBee, el cual es el mismo que se seleccionó para el caso de los dispositivos terminales (XBee-PRO), con la diferencia que en este caso funcionará como un dispositivo coordinador de funciones generales.

Además, se menciona que el adquirente de datos puede estar localizado en un ambiente indoor, donde el mismo pueda hacer uso de corriente alterna, así como puede estar localizado en un ambiente outdoor, donde sea necesario la utilización de paneles solares y baterías DC. En función de los ambientes en los cuales el adquirente de datos se encuentra localizado, se presentan las correspondientes características técnicas del computador en la Tabla 2.7. Se debe destacar que los sensores siempre se localizarán en un ambiente outdoor, en los correspondientes puntos de medición.

**Tabla 2.7 Características Técnicas para Ambientes Indoor y Outdoor del Adquirente de Datos**

<b>Características Técnicas</b>	<b>Computador de Ambiente Indoor</b>	<b>Panel TPC-1071H de Advantech</b>
Memoria RAM	512 MB	4 GB
Puertos seriales RS232	-----	2
Puertos USB	2	2
Energización	110VAC	10-32 VDC
Consumo de energía	-----	17W
Rango de humedad que puede soportar	-----	0-50 °C
Rango de temperatura que puede soportar	-----	10 – 95% RH

*Fuente: [20]*

### 2.4.4. DISPOSITIVOS Y CARACTERÍSTICAS TÉCNICAS DEL MÓDULO DE TRANSMISIÓN Y PROCESAMIENTOS DE DATOS GPRS

Se recuerda que este módulo implementa la solución GPRS ya utilizada y en funcionamiento por el INAMHI. En este sentido, se presenta en la Tabla 2.8 las características técnicas de los dispositivos y equipos necesarios.

Se debe denotar que la conexión punto a punto GPRS, se realiza por medio de un cliente, desempeñado por el módulo de adquisición de datos, y un servidor localizado en la estación de INAMHI Iñaquito. El modem utilizado se observa en la Figura 2.16.

**Tabla 2.8 Características Técnicas del Equipos y Dispositivos del Módulo de Transmisión y Procesamiento de Datos GPRS**

Equipos y Dispositivos del Módulo de Transmisión y Procesamiento de Datos GPRS	Características Técnicas
Modem GPRS Fastrack Wavecom	Voltaje de alimentación: 12VDC Interfaz serial RS232 Sólido para tarjeta SIM Frecuencia de operación: 850 / 900 / 1800 / 1900 MHz 3.5 dB de ganancia de antena omnidireccional
Fuente de Alimentación	12VDC
Servidor de Datos Meteorológicos	CPU de 1GHz Memoria RAM de 1GB 500 GB de almacenamiento Sistema operativo XP/7

Fuente: [21]



**Figura 2.16 Modem de Transmisión GPRS Fastrack Wavecom**

Fuente: [21]

#### 2.4.5. REQUERIMIENTOS DE SOFTWARE PARA IMPLEMENTACIÓN DEL PROTOTIPO

Se selecciona el software, en base al dispositivo que se necesite programar o configurar, de acuerdo a las funciones de los distintos módulos. En este sentido, se especifica las siguientes aplicaciones:

- a) AVR Studio 6.0 de Atmel, para la programación del micro-controlador ATmega164P. Atmel Studio es un entorno para el desarrollo AVR y ARM, basado en programación C++.

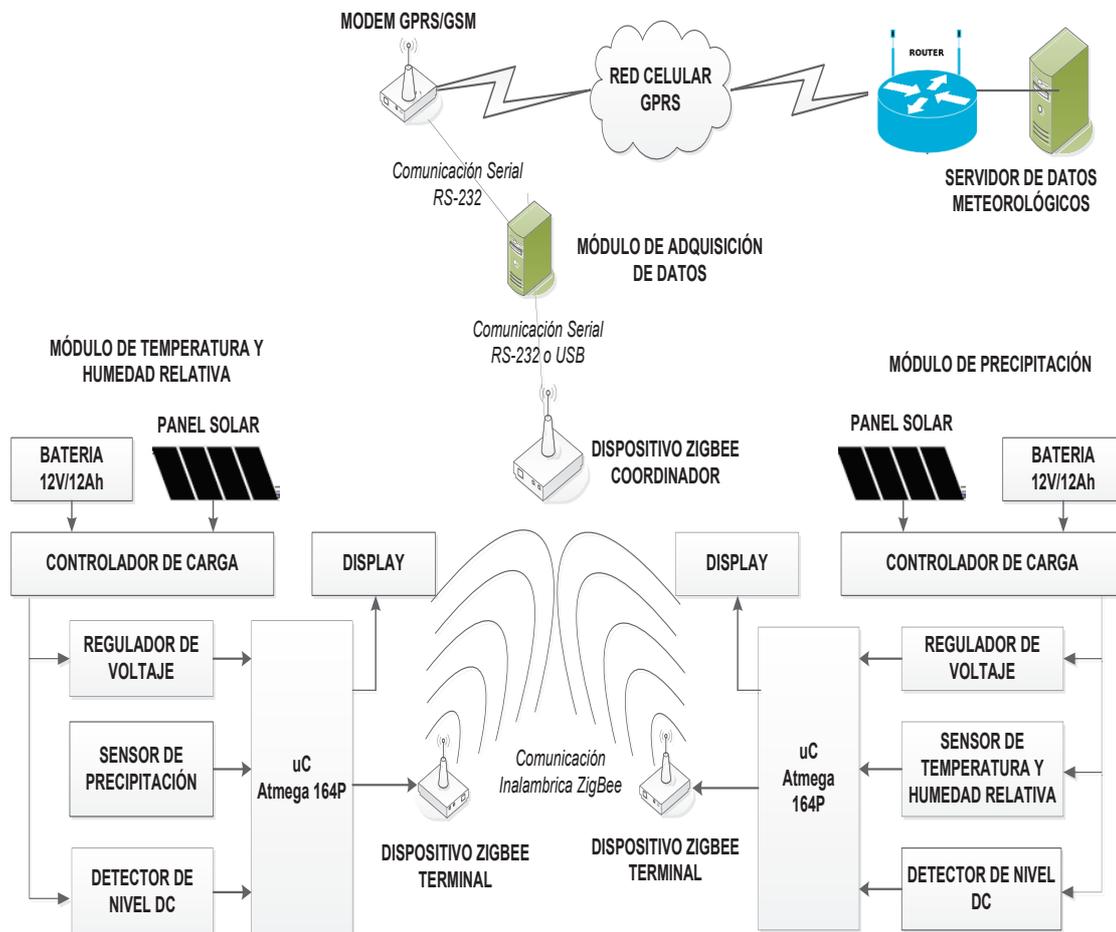
- b) X-CTU del fabricante Digi, para programación y configuración de los módems RF XBee-PRO. X-CTU es una aplicación para Windows diseñada para interactuar con los archivos del firmware de los dispositivos RF mediante un interfaz gráfico, facilitando así la utilización de comandos AT.
- c) LabVIEW 2010 de National Instruments, para desarrollo del programa que permita la adquisición, procesamiento y presentación de datos meteorológicos. LabVIEW es una plataforma y entorno de desarrollo para diseño de sistemas de control y pruebas, basado en programación gráfica.
- d) Altium Designer, para desarrollo de las placas y PBCs de los módulos de precipitación, temperatura y humedad relativa. Este software permite el desarrollo de placas de circuitos impresos, FPGA y el diseño de software embebido.
- e) Open AT del fabricante Sierra Wireless, para configuración y programación del modem Fastrack Wavecom. Open AT es la manera de integrar y ejecutar la aplicación de software basada en C++ en el modem RF, para el desarrollo de aplicaciones M2M.

## CAPÍTULO 3

### DISEÑO DEL PROTOTIPO

#### 3.1. PROCEDIMIENTO DE DISEÑO DEL PROTOTIPO

El procedimiento para diseño del prototipo se realiza de acuerdo a los requerimientos y funciones de cada uno de los módulos del prototipo previamente planteados en el Capítulo 2. De esta forma, tomando en cuenta el esquema de la Figura 3.1 a ser implementado, el procedimiento para el diseño debe ser realizado de acuerdo a los siguientes pasos:



**Figura 3.1** Esquema del Prototipo Inalámbrico utilizando el Estándar ZigBee y Transmisión GPRS para el Monitoreo y Adquisición de Datos Meteorológicos

*Fuente: Elaboración propia en base a los requerimientos del prototipo planteados en el capítulo 2*

- 1) Especificar los sistemas y elementos de cada uno de los módulos del prototipo.

- 2) Establecer diagramas de flujo o de bloques, que definan procedimientos que permitan cumplir con las funciones previamente definidas por cada uno de los elementos y sistemas de los módulos del prototipo.
- 3) Diseño de programas y circuitos en base a los diagramas de flujo y de bloques planteados para el funcionamiento de cada uno de los módulos del prototipo.
- 4) En base a detección de errores y modificaciones en el diseño de programas y circuitos, establecer una realimentación de información para mejorar el diseño total del prototipo.

### **3.2. DISEÑO DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA**

En esta sección se especifican diagramas de flujo, diagramas de bloque, estructuras de programas, circuitos y procedimientos, que permitan el diseño de cada uno de los sistemas y elementos del módulo de temperatura y humedad relativa.

#### **3.2.1. SISTEMAS Y ELEMENTOS DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA**

Tomando en cuenta el diagrama de la Figura 3.2, los sistemas del módulo de temperatura y humedad relativa son:

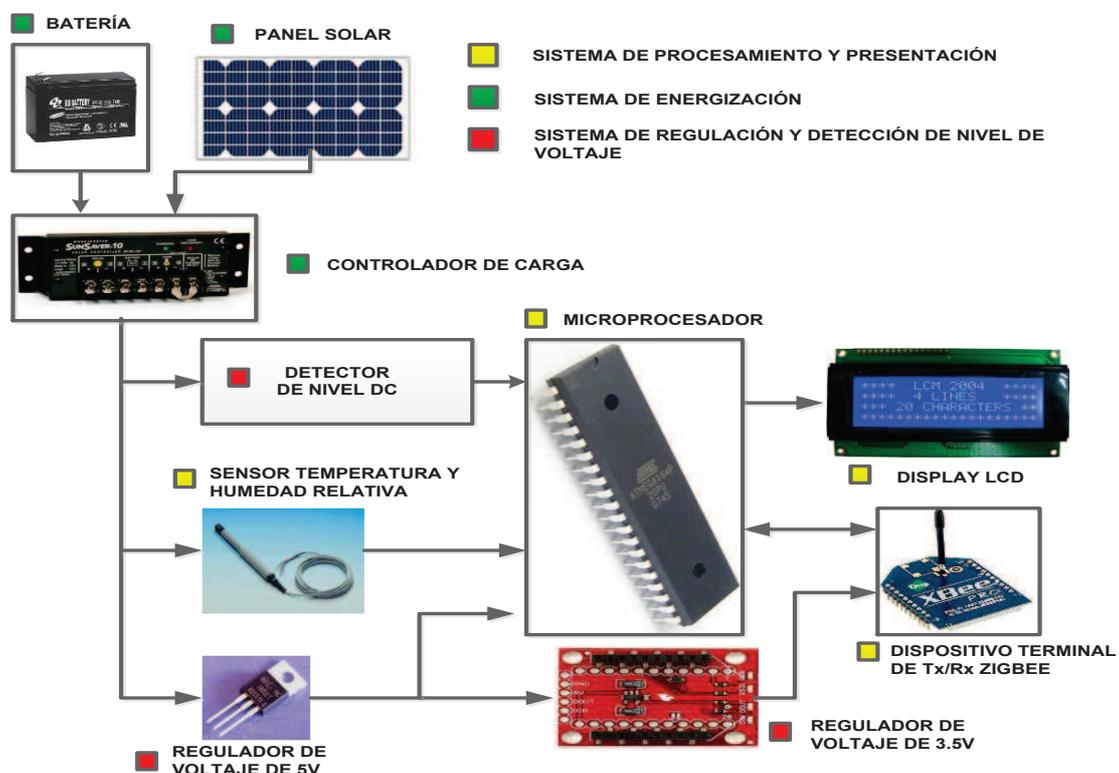
- a) Sistema de Procesamiento y Presentación.
- b) Sistema de Energización.
- c) Sistema de Regulación y Detección de Nivel de Voltaje.

#### **3.2.2. DESCRIPCIÓN Y DISEÑO DEL SISTEMA DE PROCESAMIENTO Y PRESENTACIÓN DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA**

El sistema de procesamiento y presentación está conformado por el microprocesador ATmega164P, el dispositivo de Tx/Rx ZigBee, un display LCD de 4x20 caracteres y el sensor de temperatura y humedad relativa HMP45A. En este sentido las funciones que debe desempeñar cada uno de los elementos de este sistema son:

## Microprocesador ATmega164P

- Conversión analógica-digital de las señales de voltaje del sensor de temperatura y humedad relativa y del detector de nivel DC, por medio del subsistema ADC que presenta el microprocesador ATmega164P.
- Muestreo de la conversión analógica-digital en un intervalo de tiempo determinado, mediante la utilización del temporizador y contador Timer/Counter1 de 16 bits del microprocesador ATmega164P.
- Transmisión asincrónica de los datos digitales de temperatura y humedad relativa al dispositivo de Tx/Rx ZigBee, por medio del subsistema UART que presenta el microprocesador ATmega164P.
- Presentación de datos de nivel de voltaje necesario para funcionamiento del módulo, datos de temperatura y humedad relativa, mediante la utilización de las funciones de los pórtilos de entrada/salida digitales de propósito general del ATmega164P, a los cuales se conecta el display LCD.
- Detección de la interrupción externa INT1, que permita el encendido de la pantalla LCD.



**Figura 3.2 Diagrama de Bloques del Módulo de Temperatura y Humedad Relativa**

*Fuente: Elaboración propia basada en la Figura 3.1*

## Dispositivo de Tx/Rx ZigBee XBee-PRO

- Operación como dispositivo terminal de funciones reducidas (*Reduced Function Device* RFD), utilizando el programa X-CTU para su correspondiente configuración.
- Recepción asincrónica de la información digital en caracteres de texto por parte del microprocesador, mediante la utilización del interfaz UART.
- Transmisión inalámbrica de tramas API al dispositivo de Tx/Rx ZigBee Coordinador, de la información procesada del microprocesador mediante la correspondiente configuración de dicho dispositivo.

## Display LCD

- Mostrar la información entregada por el microprocesador ATmega164P.

Una vez que se han especificado las funciones que se deben desempeñar en este sistema, se procede a estructurar subprogramas que van a ser definidos como librerías dentro de un solo programa, que permitan cumplir con el correspondiente procesamiento para el caso del microprocesador, tal como se muestra en la Figura 3.3 (a), y para el caso del dispositivo de Tx/Rx XBee-PRO, se definen parámetros de configuración, tal como se muestra en la Figura 3.3 (b).

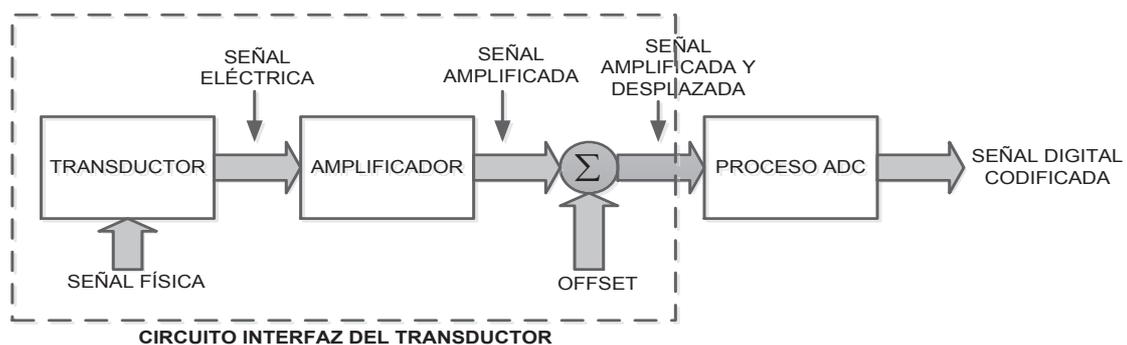
Programa de Procesamiento de Información del Módulo de Temperatura y Humedad Relativa	ADC.h	Obtención de datos digitales	Procedimiento de Configuración de los Dispositivos de Tx/Rx ZigBee FFD y RFD	Configuración del interfaz serial UART
	muestreo.h	Conversión ADC en intervalos de tiempo definidos		Configuración del interfaz RF
	UART.h	Transmisión asincrónica utilizando el interfaz UART		Configuración de asociación y topología de red del dispositivo coordinador FFD
	LCD.h	Manejo de Comandos del LCD		Configuración de asociación y topología de red del dispositivo final RFD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2		Configuración del direccionamiento de origen y destino de cada dispositivo ZigBee
				Configuración del método de acceso al medio MAC

**Figura 3.3** Esquemas que muestran: (a) Conjunto de Programas del Módulo de Temperatura y Humedad Relativa y (b) Conjunto de Parámetros a ser Configurados en el Módulo XBee-PRO RFD

Fuente: Elaboración propia

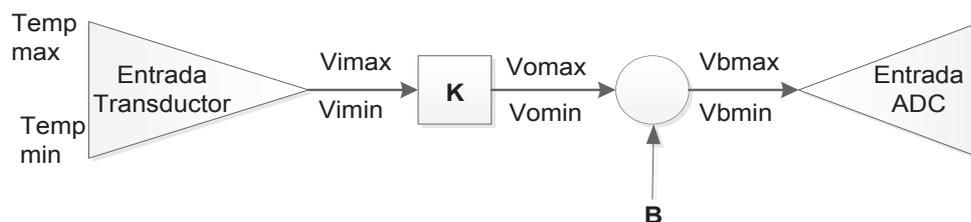
### 3.2.2.1. Conversión Analógica-Digital de las Señales del Sensor de Temperatura y Humedad Relativa [22]

El proceso de conversión analógica a digital se presenta en el diagrama de bloques de la Figura 3.4. Los transductores tienen como fin el de convertir una señal física en una señal eléctrica o viceversa, en este caso el transductor utilizado sería el sensor de temperatura y humedad relativa. También se puede observar la necesidad de un circuito de acondicionamiento de señal antes de aplicar el proceso ADC. El circuito de acondicionamiento de señal es denominado interfaz del transductor. El objetivo del circuito interfaz del transductor es de amplificación y desplazamiento del rango de la señal eléctrica para su correspondiente acoplamiento en la entrada del subsistema ADC del microprocesador. La Figura 3.5 muestra el circuito interfaz del transductor.



**Figura 3.4 Diagrama de Bloques de la Conversión ADC**

*Fuente: Elaboración propia basada en [22]*



**Figura 3.5 Diagrama de Bloques de una Señal Acondicionada para la correspondiente Conversión ADC**

*Fuente: [22]*

El proceso ADC se presenta a continuación del circuito de acoplamiento de señal, y empieza con el subproceso de muestreo de la señal analógica, tomando muy en cuenta el tipo de señal que va ser convertida para elegir correctamente la velocidad de muestreo. Por ejemplo, si se necesita convertir una señal de voz, se necesita al menos una velocidad de muestreo de 8-KHz.

Una vez que la señal haya sido muestreada, el subproceso de cuantificación toma lugar, tomando muy en cuenta el rango de error de cuantificación permitido. Para determinar el número de bits y su correspondiente error máximo de cuantificación, se debe usar la siguiente ecuación:

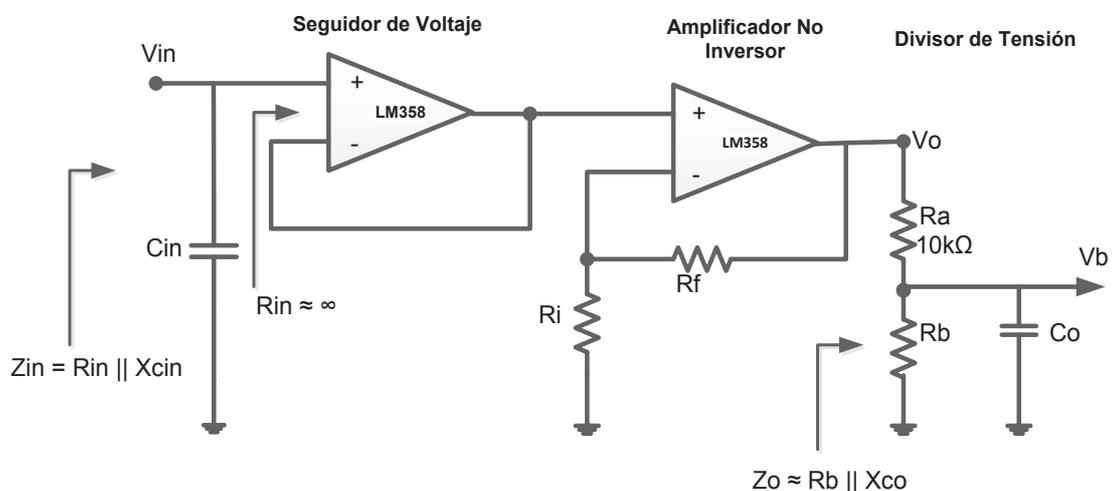
$$\text{Resolución} = \frac{\text{rango}}{2^b} \quad \text{(Ecuación 3.1)}$$

Finalmente una vez que el subproceso de cuantificación haya culminado se debe decodificar mediante un proceso de conversión de números decimales a notación binaria, considerando que el número binario hace uso de todos los bits disponibles. Por ejemplo la cuantificación del nivel 4 usando 8-bits es codificado como 00000100 y no en 100.

En resumen el proceso ADC tiene tres pasos: muestreo, cuantificación y codificación.

### 3.2.2.1.1. Diseño del Circuito Interfaz del Transductor

El diseño del circuito interfaz del transductor mostrado en la Figura 3.6, se lo realiza con el fin de acoplar las señales de entrada de temperatura y humedad, a las entradas ADC del microprocesador, mediante el uso del amplificador operacional LM358, el cual puede ser alimentado con 5 V. Por tanto, se procede a describir cada uno de los elementos de este circuito a continuación:



**Figura 3.6 Circuito Interfaz del Transductor.**  
Fuente: Elaboración propia.

- **Seguidor de voltaje**, permite anular los efectos de carga aislando la señal de salida con respecto a la señal de entrada, debido a la alta impedancia de entrada que presenta el amplificador operacional [23]. De igual forma, se incluye un capacitor de entrada que permite reducir la interferencia de ruido que puede existir en el voltaje de entrada.

$$Z_{in} = X_{cin} \parallel R_{in} = X_{cin} \parallel \infty = X_{cin} \quad \text{(Ecuación 3.2)}$$

En consecuencia, el capacitor de entrada actúa como un filtro pasa bajas, con una frecuencia de corte de 1 KHz, siempre y cuando el valor de  $X_{cin}$  sea mucho menor que la impedancia de entrada del amplificador operacional. En ese sentido se asume  $X_{cin} = 8$ , valor que es mucho menor que  $R_{in} = \infty$  y que a su vez permite obtener un valor de capacitancia estándar.

$$X_{cin} = \frac{1}{\omega C_{in}} = \frac{1}{2\pi f C_{in}} \quad \text{(Ecuación 3.3)}$$

$$C_{in} = \frac{1}{2\pi f X_{cin}} = \frac{1}{2\pi \times 1k \times 8} = 19.9\mu F$$

$$C_{in} \approx 22\mu F$$

**Amplificador no inversor**, el cual es utilizado para obtener una ganancia de voltaje positiva, y de esta manera amplificar la señal obtenida del sensor de temperatura y humedad relativa y reducir el ruido obtenida de dicha señal [23]. En este sentido, se procede con el cálculo de los valores de las correspondientes resistencias, tomando en cuenta que el voltaje de entrada máximo a ser amplificado es de 1 V, el cual representa el valor máximo que el sensor de temperatura o humedad puede censar. El valor de amplificación fue obtenido en base a mediciones empíricas, tomando en cuenta que el voltaje de entrada máximo que un canal ADC soporta es de 2.5 V y la alimentación del circuito es de 5 V. En este sentido, se obtuvo un valor de 3.7, el cual representa el valor máximo de ganancia que el circuito integrado LM358 puede amplificar, cuando se encuentra alimentado por una sola fuente de 5V. Sin embargo, se debe mencionar que una ganancia de 3.7 es suficiente para amplificar valores de voltaje del sensor de temperatura y humedad que presentan un rango de variación de 0 V a 1 V.

**Datos:**

$$V_{imax} = 1V$$

$$V_{imin} = 0V$$

$$K = 3.7$$

$$K = \frac{V_{omax}}{V_{imax}} \quad \text{(Ecuación 3.4)}$$

$$V_{omax} = K \times V_{imax} = 3.7 \times 1V$$

$$V_{omax} = 3.7V$$

Determinar el  $R_i$  y  $R_f$  del amplificador no inversor:

$$V_o = \left(1 + \frac{R_f}{R_i}\right) V_{in} \quad \text{(Ecuación 3.5)}$$

$$R_f = \left(\frac{V_o}{V_{in}} - 1\right) R_i = (K - 1) R_i = (3.7 - 1) R_i$$

$$R_f = 2.7 R_i$$

Se asume que  $R_i = 100k\Omega$ , entonces  $R_f = 270k\Omega$ .

- **Divisor de tensión**, para acoplar la señal de voltaje amplificada al rango de voltaje que el microcontrolador opera en sus entradas digitales. Para el caso del ATmega164P, el rango de voltaje de operación de las entradas ADC es de 0 V-2,55 V cuando se utiliza un voltaje de referencia interno. De esta manera, se procede a determinar las correspondientes resistencias tomando en cuenta que el voltaje máximo a ser amplificado es de 3.7V.

**Datos:**

$$V_{omax} = 3.7V$$

$$V_{bmax} = 2.55V$$

$$V_{bmin} = 0V$$

$$V_b = \left(\frac{R_b}{R_a + R_b}\right) V_o \quad \text{(Ecuación 3.6)}$$

$$R_b = (R_a + R_b) \frac{V_b}{V_o}$$

$$\left(1 - \frac{V_b}{V_o}\right) R_b = \frac{V_b}{V_o} R_a$$

$$R_b = \frac{\frac{V_b}{V_o} R_a}{\left(1 - \frac{V_b}{V_o}\right)} = \frac{\frac{2.55}{3.7} R_a}{\left(1 - \frac{2.55}{3.7}\right)} R_a = 2.21739 R_a$$

Se asume que  $R_a = 10 \text{ k}\Omega$ , entonces  $R_b = 22.17 \text{ k}\Omega \approx 22 \text{ k}\Omega$ .

$$Z_o = X_{co} \parallel R_b = X_{co} \parallel 22 \text{ k}\Omega = X_{co} \quad (\text{Ecuación 3.7})$$

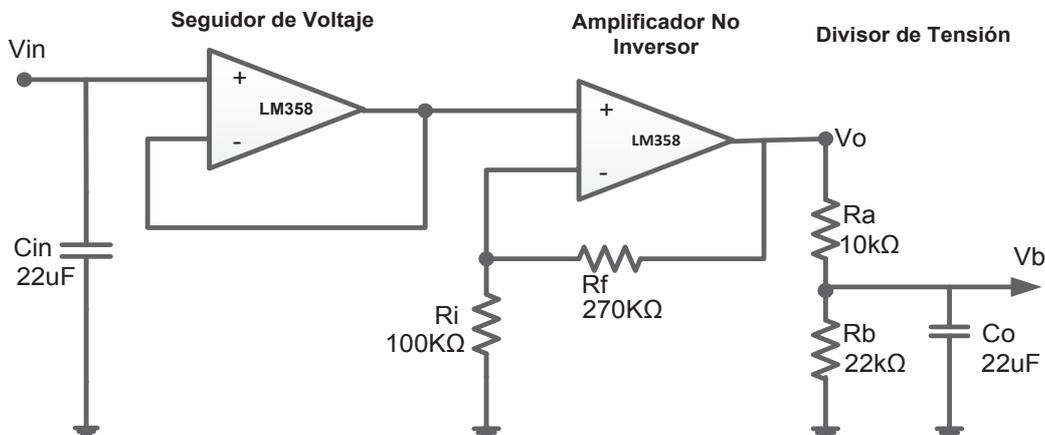
De igual manera, el capacitor de salida actúa como un filtro pasa bajas, con una frecuencia de corte de 1 KHz, para reducir interferencias de ruido en la entrada de un canal ADC, siempre y cuando el valor de  $X_{co}$  sea mucho menor que  $R_b$  previamente calculado. Por lo tanto, se asume  $X_{co} = 8$ , valor que es mucho menor que  $R_b = 22 \text{ k}\Omega$  y que a su vez permite obtener un valor de capacitancia estándar.

$$X_{co} = \frac{1}{\omega C_o} = \frac{1}{2\pi f C_o} \quad (\text{Ecuación 3.8})$$

$$C_o = \frac{1}{2\pi f X_{co}} = \frac{1}{2\pi \times 1\text{k} \times 8} = 19.9 \mu\text{F}$$

$$C_o \approx 22 \mu\text{F}$$

La Figura 3.7 muestra los valores de las resistencias y capacitores determinados previamente.



**Figura 3.7 Valores de Resistencias del Circuito Interfaz del Transductor.**  
Fuente: Elaboración propia.

### 3.2.2.1.2. Configuración y Manejo del ADC del ATmega164P

Se debe mencionar que el programa de conversión ADC<sup>1</sup> permite obtener información digital en caracteres de texto ASCII de las señales de temperatura y humedad relativa. De igual manera, este programa va ser utilizado como una librería (ADC.h) del programa principal del sistema de procesamiento, tal como se muestra en la Figura 3.8.

Programa de Procesamiento de Información del Módulo de Temperatura y Humedad Relativa	ADC.h	Obtención de datos digitales
	muestreo.h	Conversión ADC en intervalos de tiempo definidos
	UART.h	Transmisión asincrónica utilizando el interfaz UART
	LCD.h	Manejo de Comandos del LCD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2

**Figura 3.8** Librería ADC.h dentro del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa

Fuente: Elaboración propia

Las funciones que se definen en la librería ADC.h son:

1) Función de inicialización de conversión ADC simple (inic\_ADC\_simple), en la cual se configuran los siguientes parámetros:

- Habilitación del ADC.

$$ADC = \frac{V_{IN} \times 1024}{V_{REF}} \quad \text{(Ecuación 3.9)}$$

- La Tabla 3.1 muestra las opciones de fuente de conversión ADC que presenta el microprocesador Atmega164P.
- Establecimiento de un valor de pre-escalamiento de 8 para obtener una frecuencia de reloj para operación del ADC (clkADC) entre 50KHz y 200kHz.

<sup>1</sup> En el Anexo A.2 se muestra la librería adc.h del módulo de temperatura y humedad relativa, correspondiente a la lectura de tres canales de conversión ADC.

- Establecimiento de un voltaje de referencia interno de 2.56V.
- Habilitación del ajuste de resultado en la derecha.
- Des-habilitación de los canales ADC que no se van a utilizar (AD3-AD7).

**Tabla 3.1 Selección de la Fuente de Auto Activación ADC**

ADTS2	ADTS1	ADTS0	Fuente a ser Activada
0	0	0	Modo de libre corrimiento
0	0	1	Comparador analógico
0	1	0	Petición de interrupción externa 0
0	1	1	Detección de comparación del Timer/Counter0
1	0	0	Desbordamiento del Timer/Counter 0
1	0	1	Detección de comparación del Timer/Counter 1 B
1	1	0	Desbordamiento del Timer/Counter 1
1	1	1	Captura de un evento del Timer/Counter 1

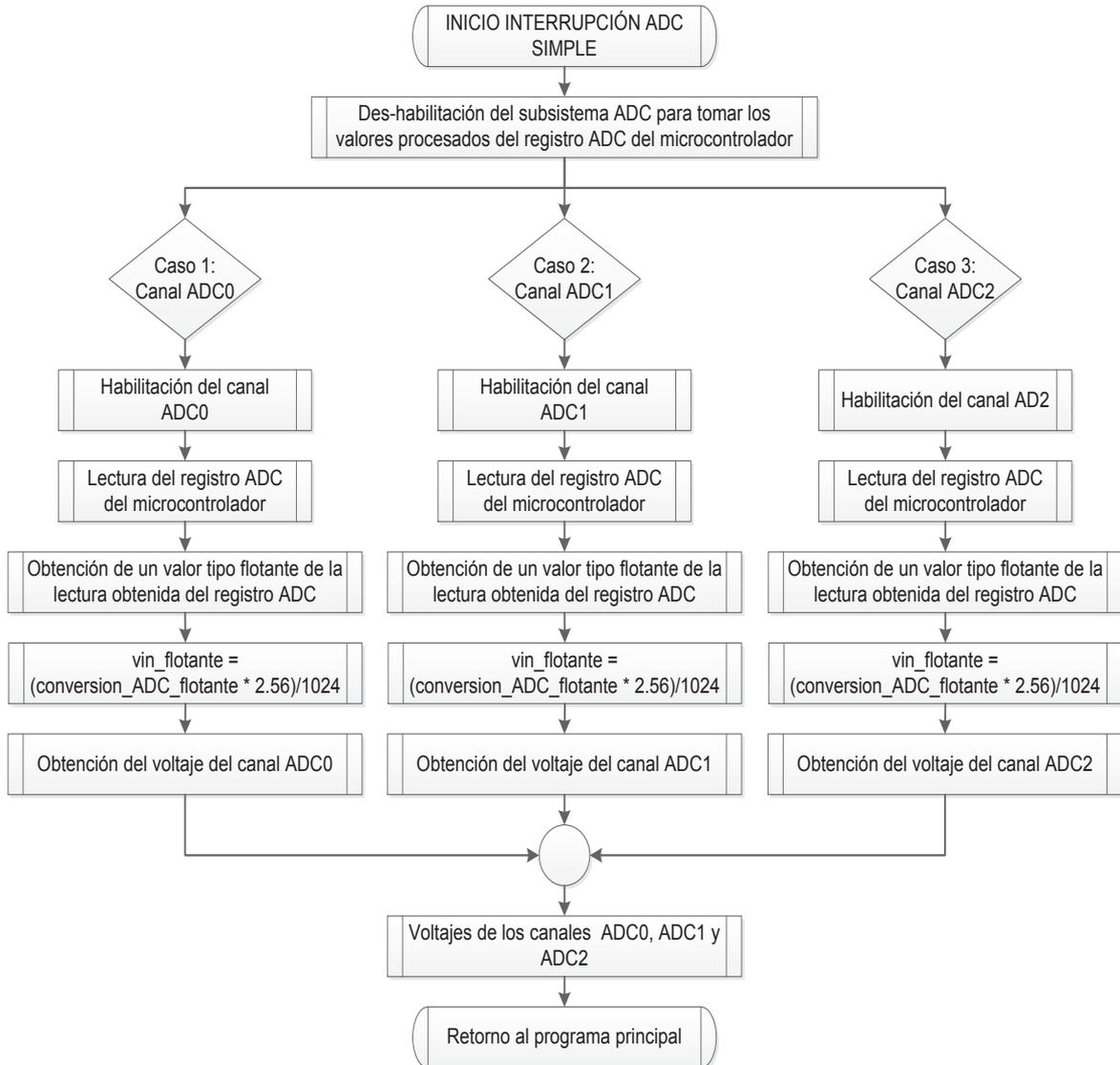
*Fuente: Elaboración propia basado en [24]*

2) Función de interrupción ADC, la cual se ejecuta cuando se realiza una conversión analógica digital en cualquiera de los canales AD0, AD1 o AD2 .El diagrama de flujo de la Figura 3.9, muestra la ejecución de esta interrupción.

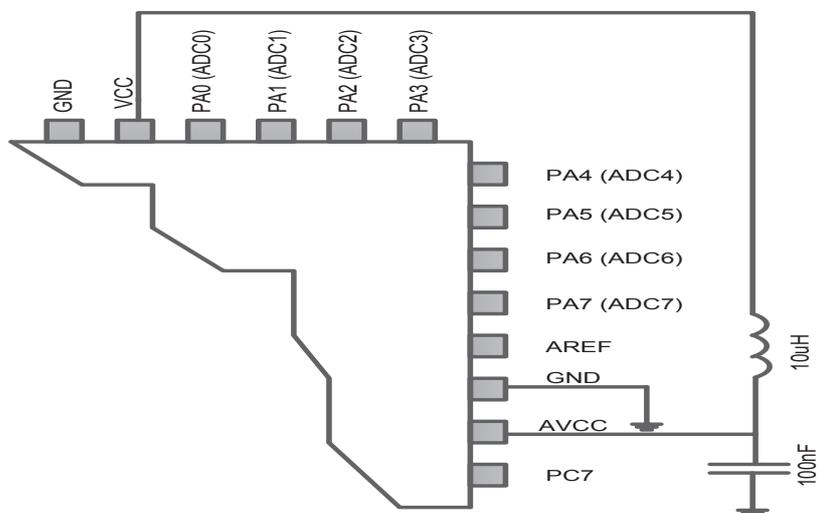
#### 3.2.2.1.3. Técnicas de Cancelación del Ruido Analógico

Circuitos digitales dentro y fuera del dispositivo generan interferencias electromagnéticas, lo cual puede afectar la precisión de las medidas analógicas. Si la precisión de conversión es crítica, el nivel de ruido puede ser reducido aplicando las siguientes técnicas:

- a) Mantener las rutas de la señal analógica tan cortas como sea posible. Se debe asegurar de que las pistas analógicas se extiendan sobre el plano de tierra analógica, y mantenerlos alejados de las pistas de alta velocidad de comunicación digital.
- b) El pin AVCC sobre el dispositivo debe ser conectado al voltaje de suministro VCC digital vía una red LC como se muestra en la Figura 3.10.
- c) Si algún pin del pósito ADC es usado como salida digital, es esencial que este no cambie mientras se encuentre en progreso una conversión.



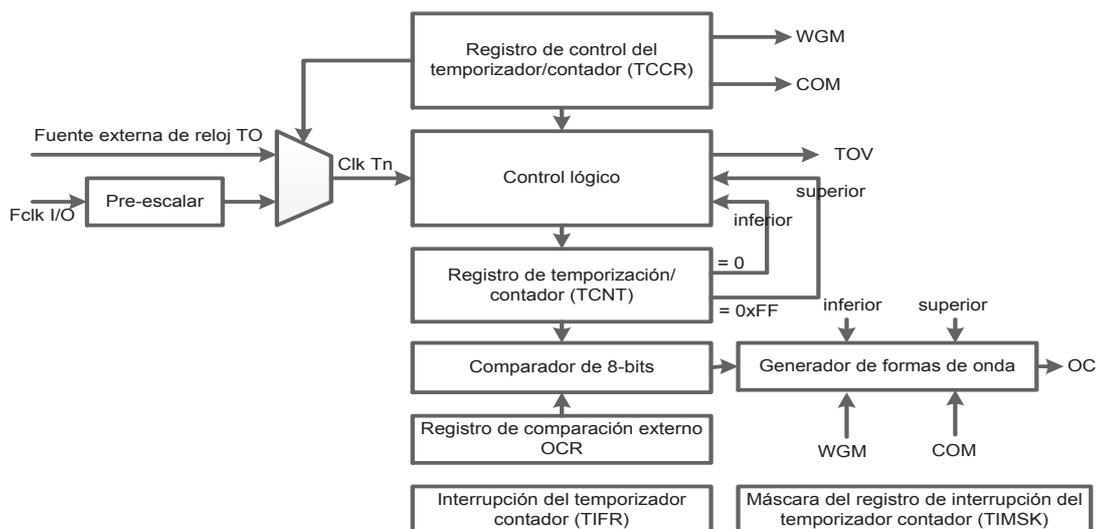
**Figura 3.9 Diagrama de Flujo de la Función de Interrupción ADC ISR(ADC\_vect) .**  
 Fuente: Elaboración propia.



**Figura 3.10 Conexión del Pin AVCC del ADC para Reducción del Ruido.**  
 Fuente: [24]

### 3.2.2.2. Temporización y Muestreo de la Conversión Analógica-Digital ADC [25]

El corazón del sistema de temporización es la base de tiempo del cristal. La frecuencia de cristal de un microprocesador es usada para generar una señal de reloj base. Para un sistema de temporización, tal como se observa en la Figura 3.11, el sistema de reloj es usado para actualizar los contenidos de un registro especial denominado contador de libre corrimiento, el cual incrementa su valor cada vez que se detecta una transición positiva o negativa de una señal de reloj. Por ejemplo, si un reloj tiene una velocidad de 2MHz, el contador de libre corrimiento incrementará su valor cada 0,5us. Todas las demás unidades de referencias de tiempo relacionadas con el contenido del contador de libre corrimiento, se utilizan para el manejo de las actividades relacionadas con la temporización de las entradas y salidas digitales (I/O) como: medida de periodos, captura de eventos de temporización y generación de señales de reloj.



**Figura 3.11 Diagrama de Bloques de un Temporizador/Contador**

Fuente: [25]

#### 3.2.2.2.1. Configuración y Manejo del Timer/Counter1 del ATmega164P

Se debe mencionar que el programa de temporización y muestreo ADC<sup>2</sup>, permite realizar la conversión ADC en intervalos de tiempo definidos por el temporizador

<sup>2</sup> En el Anexo A.4 se muestra el desarrollo de la librería muestreo.h, para la obtención de muestras de los valores de conversión ADC cada segundo, por medio de la utilización de temporizadores.

Timer/Counter1. De igual manera este programa va ser utilizado como una librería (muestreo.h) del programa principal del sistema de procesamiento, tal como se muestra en la Figura 3.12.

Programa de Procesamiento de Información del Módulo de Temperatura y Humedad Relativa	ADC.h	Obtención de datos digitales
	muestreo.h	Conversión ADC en intervalos de tiempo definidos
	UART.h	Transmisión asincrónica utilizando el interfaz UART
	LCD.h	Manejo de Comandos del LCD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2

**Figura 3.12** Librería muestreo.h dentro del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa

*Fuente: Elaboración propia.*

La configuración del subsistema de Temporización y Conteo 1 (Timer/Counter1) del microprocesador ATmega164P, se presenta de acuerdo a las funciones definidas en la librería muestreo.h, las cuales son:

- 1) Función de inicialización del Timer/Counter 1 (inic\_timer\_counter\_1), en la cual se definen los siguientes parámetros:
  - Valor pre-escalamiento del reloj del temporizador en 64.
  - Inicialización del registro de conteo en cero (CTC), cuando se detecte que la comparación del registro OCR1A/B es igual al del contador, tal como se muestra en la Tabla 3.2.

**Tabla 3.2** Descripción de los Modos de Generación de Formas de Onda Requeridos.

WGM13	WGM12 (CTCn)	WGM11 (PWM11)	WGM10 (PWM10)	Modo de Operación del Timer/Counter	Limite Superior	Actualización del OCR1A/B	Activación de la Bandera TOVn en
0	0	0	0	Normal	0xFFFF	Inmediato	Valor máximo
0	1	0	0	CTC	OCR1A	Inmediato	Valor máximo
1	1	0	0	CTC	ICR1	Inmediato	Valor máximo

*Fuente: Elaboración propia basado en [24]*

- Cambio de transición cuando se detecte una comparación con el registro OCR1A/B.
- Determinación del valor del registro OCR1A

$$OCR1A = \frac{T_{OC1A} \times f_{CLK_{IO}}}{2N} - 1 \quad (\text{Ecuación 3.10})$$

$$OCR1A = \frac{1\text{seg} \times 4\text{MHz}}{2 \times 64} - 1 = 31250$$

$$OCR1A = 0X7A12^3$$

- Determinación del valor del registro OCR1B

$$OCR1B = \frac{T_{OC1B} \times f_{CLK_{IO}}}{2N} - 1$$

$$OCR1B = \frac{1\text{seg} \times 4\text{MHz}}{2 \times 64} - 1 = 31250$$

$$OCR1B = 0X7A12$$

**Donde:**

OCR1/B: Registros de comparación.

$T_{OC1A/B}$ : Periodos de duración antes de que el contador se resetee en cero, e iguales a 1 segundo.

$f_{CLK_{IO}}$ : Frecuencia del reloj del temporizador de 4MHz.

N: Valor de pre-escalamiento de 64.

- 2) Función de interrupción del registro OCR1A, la cual se activa cuando se detecta que la comparación entre este registro y el contador coincide. El diagrama de flujo de la Figura 3.13(a), muestra la ejecución de esta interrupción.

---

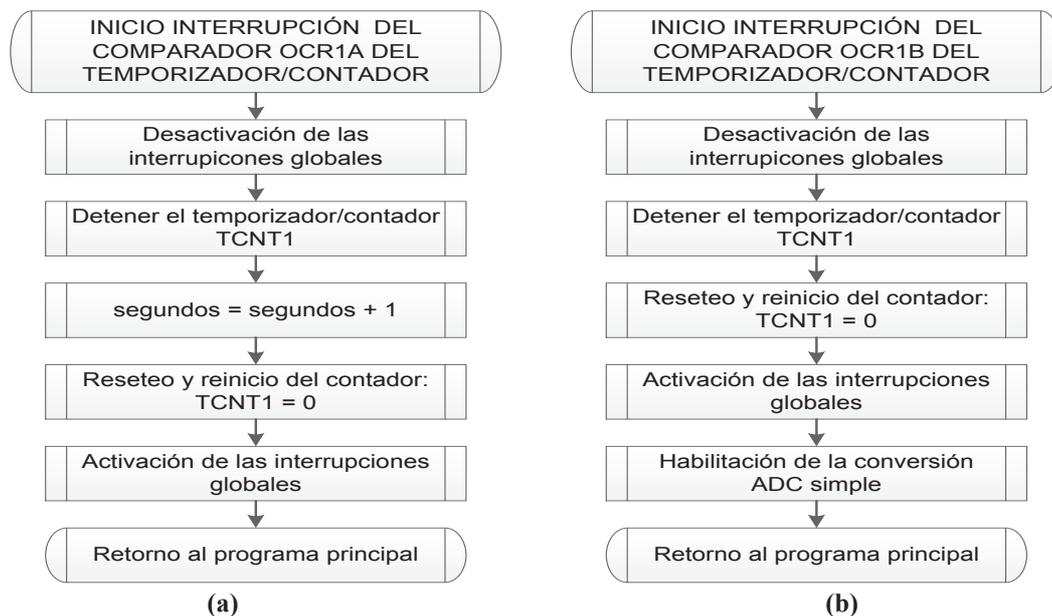
<sup>3</sup> Todos los valores después del prefijo 0X... se refieren a valores hexadecimales.

3) Función de interrupción del registro OCR1B, la cual se activa cuando se detecta que la comparación entre este registro y el contador es igual. El diagrama de flujo de la Figura 3.13(b), muestra la ejecución de esta interrupción.

### 3.2.2.3. Transmisión Asincrónica de Datos Digitales utilizando el Interfaz UART.

Una trama serial es definida como un carácter conformado por bits de datos y bits de sincronización (bits de inicio y parada), y opcionalmente un bit de paridad para chequeo de errores. En resumen:

- 1 bit de inicio.
- 5, 6, 7, 8 o 9 bits de datos.
- Ningún, par o impar bit de paridad.
- 1 o 2 bits de parada.



**Figura 3.13 Diagrama de Flujo de la Interrupción del Registro: (a) OCR1A y (b) OCR1B.**

*Fuente: Elaboración propia.*

Una trama empieza con el bit de inicio seguido por el bit menos significativo del campo de datos, luego los siguientes bits de datos, se cuentan un total de 9, en forma sucesiva terminando con el bit más significativo. Si se habilita el bit de paridad, éste se inserta después de los bits de datos, antes de los bits de parada. Cuando una trama completa es transmitida, ésta puede ser directamente seguida

por una nueva trama, o la línea de comunicación puede ser configurada a un estado alto de hibernación [24].

La Figura 3.14 muestra las posibles combinaciones de los formatos de trama. Los bits entre corchetes son opcionales.



**Figura 3.14** Formato de Tramas USART.

Fuente: [24, p. 176]

**Donde:**

- St: Bit de inicio, siempre en estado lógico bajo.
- (n): Bits de datos (0-8).
- P: bit de paridad, el cual puede ser par o impar.
- Sp: Bit de parada, siempre en nivel lógico alto.
- IDLE: Estado de reposo, en nivel lógico alto.

*3.2.2.3.1. Configuración y Manejo del Interfaz USART del ATmega164P*

Programa de Procesamiento de Información del Módulo de Temperatura y Humedad Relativa	ADC.h	Obtención de datos digitales
	muestreo.h	Conversión ADC en intervalos de tiempo definidos
	UART.h	Transmisión asincrónica utilizando el interfaz UART
	LCD.h	Manejo de Comandos del LCD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2

**Figura 3.15** Librería UART.h dentro del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa

Fuente: *Elaboración Propia.*

Se debe mencionar que el programa de comunicación serial utilizando el interfaz UART<sup>4</sup>, permite la transmisión asincrónica entre el microprocesador ATmega164P

<sup>4</sup> En el Anexo A.6 se muestra la librería `uart.h` desarrollada para el manejo y control del interfaz UART, que en el caso del módulo de temperatura y humedad relativa se utilizará tanto la transmisión como recepción de datos.

y el módulo de Tx/Rx ZigBee XBee-PRO. De igual manera este programa va ser utilizado como una librería (UART.h) del programa principal del sistema de procesamiento, tal como se muestra en la Figura 3.15.

La configuración del interfaz USART del microprocesador ATmega164P se realiza por medio de las funciones que se definen en la librería UART.h, las cuales son:

- 1) Función de inicialización del interfaz UART (inic\_UART), en la cual se definen los siguientes parámetros:
  - Modo de comunicación asincrónico UART.
  - Determinación de la velocidad de transmisión mediante la configuración de los registros de velocidad de transmisión UBRR0L y UBRR0H, de acuerdo a las especificaciones mostradas en la Tabla 3.3.

**Tabla 3.3 Ecuaciones para el Cálculo de la Velocidad de Transmisión y su Correspondiente Valor de Registro**

Modo de Operación	Ecuación para el Cálculo de la Velocidad de Transmisión	Ecuación para el Cálculo del Valor del UBRR
Modo asincrónico normal (U2X0) = 0	$BAUD = \frac{f_{osc}}{16(UBRR0 + 1)}$	$UBRR0 = \frac{f_{osc}}{16BAUD} - 1$
Modo asincrónico de velocidad doble (U2X0 = 1)	$BAUD = \frac{f_{osc}}{8(UBRR0 + 1)}$	$UBRR0 = \frac{f_{osc}}{8BAUD} - 1$
Modo sincrónico maestro	$BAUD = \frac{f_{osc}}{2(UBRR0 + 1)}$	$UBRR0 = \frac{f_{osc}}{2BAUD} - 1$

Fuente: Elaboración propia basada en [24]

**Donde:**

$f_{osc}$ : frecuencia de oscilación del reloj del sistema de 4MHz.

UBRR0: Contiene a la registros UBRRHn Y UBRRLn.

BAUD: Velocidad de transmisión de 9600bps.

$$UBRR0 = \frac{f_{osc}}{16BAUD} - 1 \quad \text{(Ecuación 3.11)}$$

$$UBRR0 = \frac{4MHz}{16 \times 9600bps} - 1 = 25 = 0X19$$

- Opción de paridad des-habilitada

- 1-bit de parada
  - 8-bits de datos
  - Habilitación del transmisor.
- 2) Función de envío de datos de temperatura al buffer Tx, cuyo objetivo es el de enviar el voltaje detectado por medio del sensor de temperatura, al buffer Tx. El diagrama de flujo de esta función se presenta en la Figura 3.16 (a).
  - 3) Función de envío de datos de humedad al buffer Tx, cuyo objetivo es el de enviar el voltaje detectado por medio del sensor de humedad, al buffer Tx. El diagrama de flujo de esta función se presenta en la Figura 3.16 (b).
  - 4) Función de envío de datos del nivel de voltaje necesario para funcionamiento del circuito al buffer Tx, cuyo objetivo es el de enviar el porcentaje de nivel de voltaje necesario para funcionamiento del circuito al buffer Tx. El diagrama de flujo de esta función se presenta en la Figura 3.16 (c).
  - 5) Función de interrupción de recepción, la cual se activa cuando se detecta contenido de datos de recepción en el buffer de Rx, y es utilizado para la lectura de los valores de temperatura y humedad procesados en el módulo de adquisición de datos. De igual manera se debe mencionar que la información recibida de temperatura y humedad respectivamente, presenta el siguiente formato: TxxxxF y HxxxxF, donde x representa los valores numéricos de estos parámetros procesados y calculados por el módulo de adquisición de datos. El diagrama de flujo de esta función se representa en la Figura 3.17.

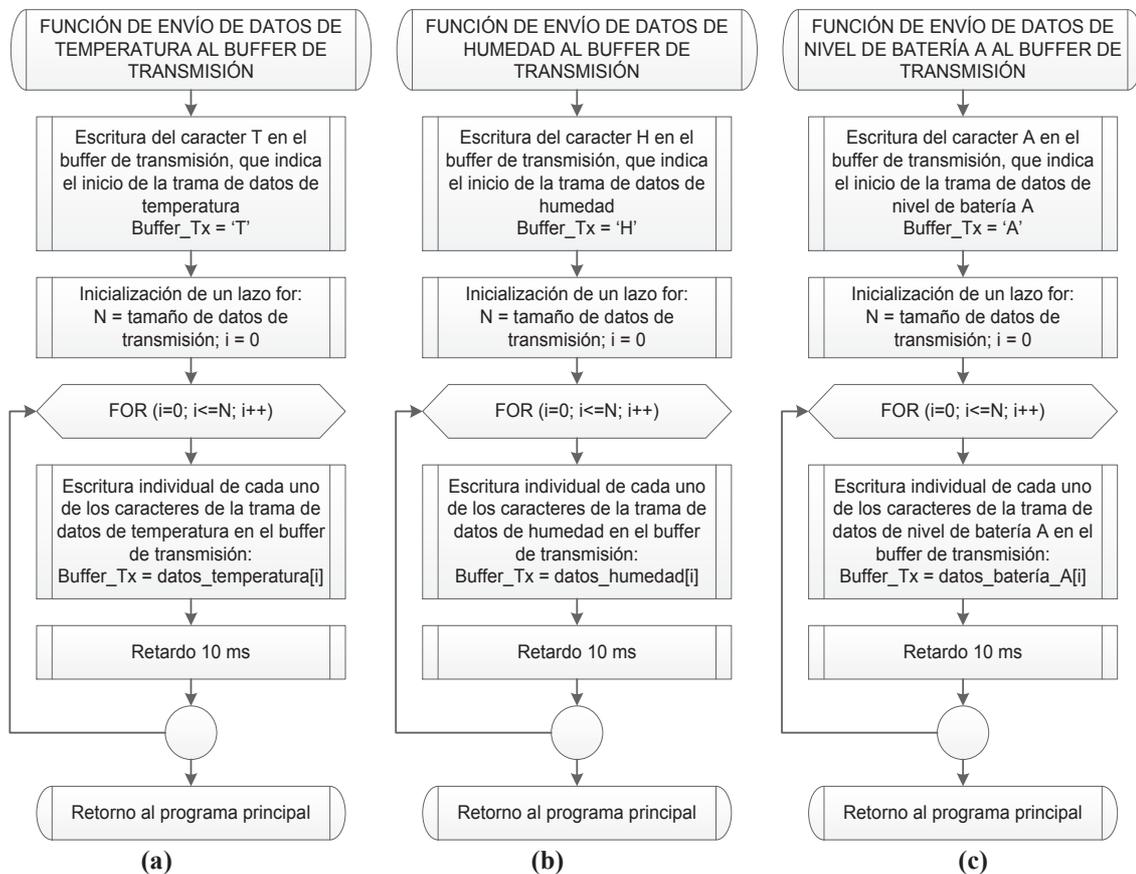
#### 3.2.2.4. Presentación de Datos Digitales en un LCD

Se debe mencionar que el programa de manejo del LCD<sup>5</sup>, permite al microprocesador ATmega164P enviar comandos al LCD por medio de la utilización de 10 pines de entrada/salida digitales I/O de propósito general. De igual manera este programa va ser utilizado como una librería (LCD.h) del

---

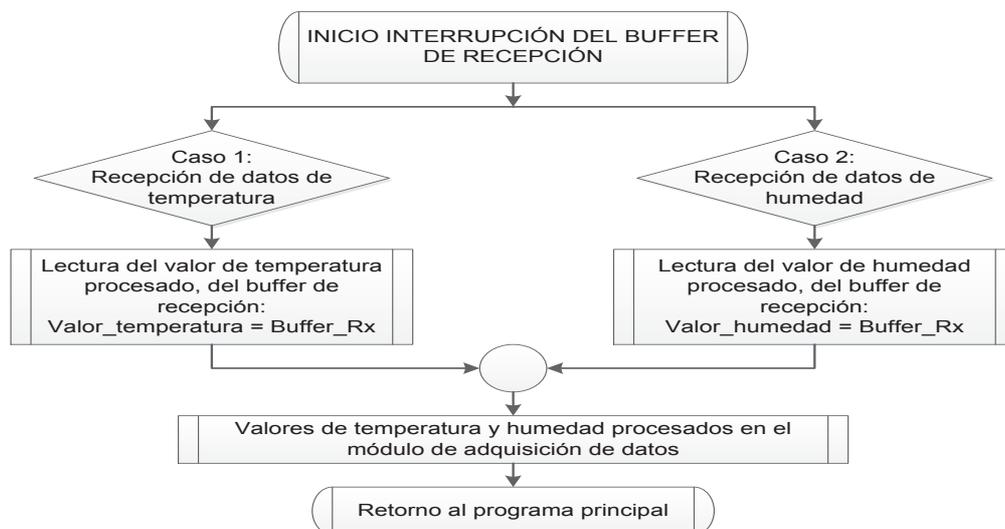
<sup>5</sup> En el Anexo A.3 se muestra el desarrollo de la librería de manejo y control de un LCD de 4x20 caracteres utilizada para los dos módulos de temperatura/humedad y precipitación.

programa principal del sistema de procesamiento, como se observa en la Figura 3.18.



**Figura 3.16 Diagramas de Flujo de las Sigüientes Funciones: (a) Envío de Datos del Sensor de Temperatura, (b) Envío de Datos del Sensor de Humedad y (c) Envío del Porcentaje de Nivel de Voltaje A para Funcionamiento del Circuito.**

*Fuente: Elaboración Propia.*



**Figura 3.17 Diagrama de Flujo de la Función de Interrupción de Recepción Activada cuando se Detecta que el Buffer Rx Contiene un Carácter de Datos.**

*Fuente: Elaboración propia.*

Programa de Procesamiento de Información del Módulo de Temperatura y Humedad Relativa	ADC.h	Obtención de datos digitales
	muestreo.h	Conversión ADC en intervalos de tiempo definidos
	UART.h	Transmisión asincrónica utilizando el interfaz UART
	LCD.h	Manejo de Comandos del LCD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2

**Figura 3.18** Librería LCD.h del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa.

*Fuente: Elaboración propia.*

#### 3.2.2.4.1. Asignación de Pines del LCD 4x20 Caracteres y Funciones de Operación

Tanto la asignación como descripción de cada uno de los pines de un LCD de 4x20 caracteres, se muestra en la Tabla 3.4.

Los 3 primeros pines son de alimentación, tomando en cuenta que el 3er pin es utilizado para establecer el contraste del LCD en función de su nivel de tensión mediante un divisor resistivo. La conexión se muestra en la Figura 3.19.

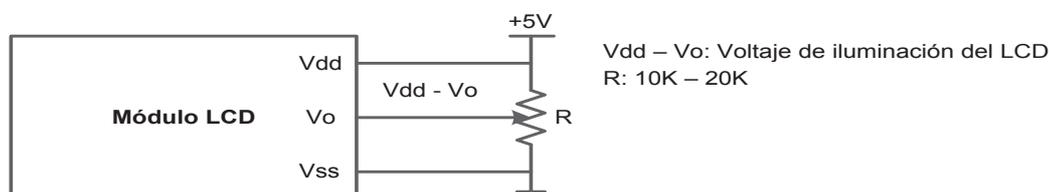
Las funciones que cumplen los pines de habilitación, lectura y escritura se muestran en la Tabla 3.5 y se establecen de acuerdo a los siguientes parámetros:

- **Pin 6 (Enable):** Habilita (1) o des-habilita (0) el módulo LCD.
- **Pin 5 (R/W):** Indica si se va a hacer una operación de escritura (0) o lectura (1).
- **Pin 4 (RS):** Selecciona el registro del LCD que se va a utilizar. El LCD tiene dos registros de 8 bits, un registro de instrucciones (0) y otro de datos (1). Seleccionando el registro de instrucciones se indica al LCD que lo que hay en el bus de datos (pines desde 7 a 14) es una instrucción, y seleccionando el registro de datos, se indica al LCD que lo que tenemos en el bus de datos es un carácter que va a ser almacenado en la memoria DDRAM o la CGRAM del LCD.

**Tabla 3.4** Tabla de Asignación de Pines de un LCD Hitachi

Pin No	Symbolo	Nivel	Descripción
1	VSS	0V	Tierra
2	VDD	5.0V	Voltaje de alimentación
3	V0	(Variable)	Voltaje de operación para el LCD
4	RS	H/L	H: Datos, L: Código de instrucción
5	R/W	H/L	H: Lectura de datos, L: Escritura de datos
6	E	H, H – L	Habilitación del LCD
7	DB0	H/L	Bit de datos 0
8	DB1	H/L	Bit de datos 1
9	DB2	H/L	Bit de datos 2
10	DB3	H/L	Bit de datos 3
11	DB4	H/L	Bit de datos 4
12	DB5	H/L	Bit de datos 5
13	DB6	H/L	Bit de datos 6
14	DB7	H/L	Bit de datos 7
15	LED(+)		Anodo de los leds de encendido del fondo de pantalla
16	LED(-)		Catodo de los leds de encendido del fondo de pantalla

Fuente: Elaboración propia basado en. [26].

**Figura 3.19** Divisor Resistivo para Manejo de Voltaje para Control del Contraste del LCD

Fuente: [26]

**Tabla 3.5** Tabla de Funciones de los Bits de Lectura, Escritura y de Registros del LCD

RS	R/W	Operaciones
0	0	Escritura de una operación interna (por ejemplo: borrado del display)
0	1	Contador de direccionamiento (DB0 – DB7)
1	0	Escritura de datos en las memorias DDRAM o CGRAM
1	1	Lectura de datos de las memorias DDRAM o CGRAM

Fuente: Elaboración propia basada en [26]

### 3.2.2.5. Detección de un Evento Externo al Microprocesador utilizando Interrupciones Externas

El proceso de una interrupción externa es parecido a una subrutina, con la diferencia de que la ejecución de la subrutina de interrupción es no anticipada por el procesador y puede ocurrir en cualquier momento.

Las interrupciones externas son realizadas en el pin INT2:0 o en cualquier pin PCINT31:0. Se debe observar que las interrupciones externas se ejecutarán aun cuando los pines PCINT31:0 o INT2:0 son configurados como salidas digitales. Esta característica permite generar una interrupción por software [24, p. 67]. Una interrupción externa puede ser generada por una transición positiva, negativa, o detección de un nivel lógico bajo en los pines correspondientes a las interrupciones externas. Las interrupciones generadas por el establecimiento de un nivel bajo o cambio de transición son detectadas de forma asincrónica. Lo cual implica que estas interrupciones pueden ser usadas para encender al AVR del estado de reposo o del modo inactivo [24].

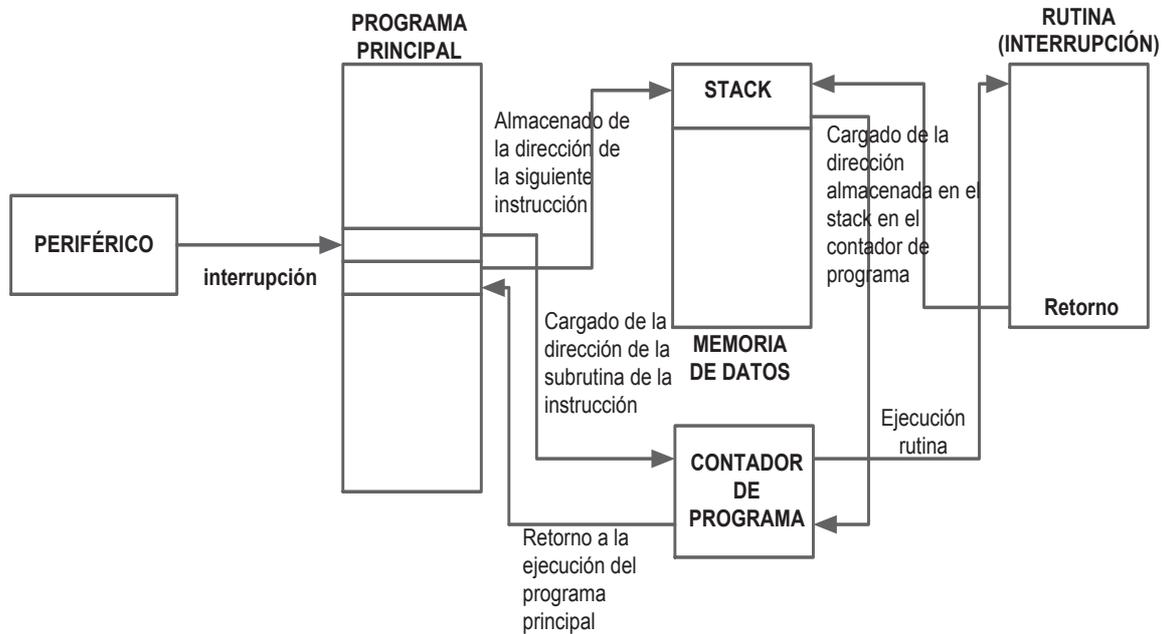
La secuencia de eventos cuando una interrupción ocurre [27], tal como se muestra en la Figura 3.20 es la siguiente:

- 1) El dispositivo periférico interrumpe al procesador, por medio del correspondiente pin de interrupción externa.
- 2) Las instrucciones en marcha terminan de ejecutarse.
- 3) La dirección de la siguiente instrucción es almacenada en la localidad stack de la memoria de datos.
- 4) La dirección del ISR (*interrupt subroutine*) es cargada en el contador de programa.
- 5) El procesador ejecuta el ISR.
- 6) La ejecución del ISR se completa con la indicación de la instrucción RETI (*return from interrupt*).
- 7) El procesador carga el contador de programa con el valor almacenado en el stack y retorna a la ejecución del programa principal o normal.

#### 3.2.2.5.1. Configuración y Manejo de Interrupciones Externas del ATmega164P

En esta sección únicamente se remitirá a la utilización de una interrupción externa para el manejo de la temporización de observación de caracteres en el LCD. Se debe mencionar que este programa va ser utilizado como una librería (*interrupcion\_externa.h*) del programa principal del sistema de procesamiento, tal como se observa en la Figura 3.21. Además, el programa de manejo de

interrupciones externas<sup>6</sup> permite detectar las basculaciones del sensor de precipitación, cuya descripción se presentará en el diseño del módulo de precipitación.



**Figura 3.20** Secuencia de Eventos del Proceso de una Interrupción Externa.

Fuente: Elaboración propia basado en [27].

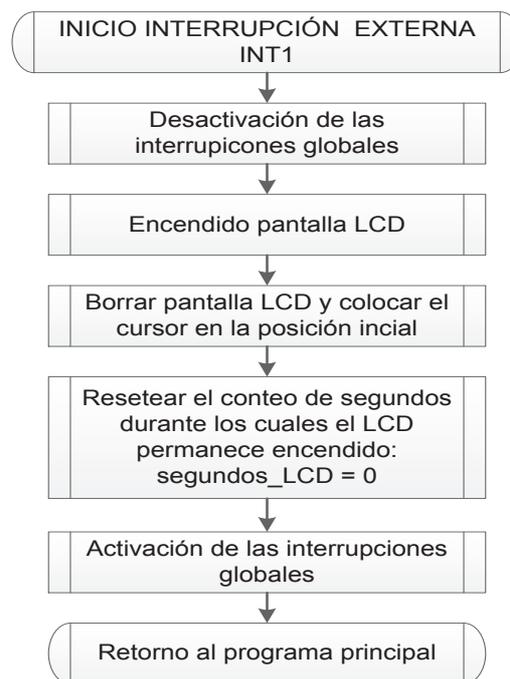
Programa de Procesamiento de Información del Módulo de Temperatura y Humedad Relativa	ADC.h	Obtención de datos digitales
	muestreo.h	Conversión ADC en intervalos de tiempo definidos
	UART.h	Transmisión asincrónica utilizando el interfaz UART
	LCD.h	Manejo de Comandos del LCD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2

**Figura 3.21** Programa de Interrupciones Externas dentro del Esquema del Conjunto de Programas del Módulo de Precipitación.

Fuente: Elaboración propia.

<sup>6</sup> El Anexo A.5, muestra la librería `interrupcion_externa.h`, la cual fue desarrollada para detección de interrupciones externas para el encendido del LCD y conteo de basculaciones del sensor de precipitación. Cabe destacar que esta librería se utiliza en los módulos de temperatura/humedad y precipitación.

Por tanto, la librería `interrupcion_externa.h` que en el microprocesador del módulo de temperatura y humedad relativa se implementa, contiene únicamente la función de interrupción externa INT1, la cual permite encender la pantalla LCD una vez que se haya detectado un cambio de transición en el pin PD3 correspondiente a esta interrupción externa. En esta subrutina también se resetea el conteo de segundos que indican cuando la pantalla del LCD se apaga tal como se muestra en el diagrama de flujo de la Figura 3.22.



**Figura 3.22 Función de Interrupción Externa INT1 para Encendido de Pantalla LCD.**

*Fuente: Elaboración Propia.*

El control de generación de interrupción externa se muestra en la Tabla 3.6, considerando que para este caso se utiliza el control de detección de una transición positiva.

**Tabla 3.6 Control de la Generación de la Interrupción Externa en el Pin INTn**

ISCn1	ISCn0	Descripción
0	0	El nivel lógico bajo de INTn genera una petición de interrupción
0	1	Cualquier cambio de transición en el pin INTn genera una petición de interrupción
1	0	La detección de una transición negativa en el pin INTn genera una petición de interrupción
1	1	La detección de una transición positiva en el pin INTn genera una petición de interrupción

*Fuente: Elaboración propia basado en [24]*

### 3.2.2.6. Programa Principal de Procesamiento de Información del Módulo de Temperatura y Humedad Relativa

El esquema del programa principal<sup>7</sup> integra las librerías de conversión ADC, muestreo, interfaz UART, manejo del LCD e interrupción externa, tal como se muestra en la Figura 3.23. Las librerías permiten realizar el procesamiento de las señales obtenidas del sensor de temperatura y humedad relativa, para a continuación enviar la correspondiente información digital al módulo de Tx/Rx ZigBee RFD. De esta forma el esquema del programa principal es el siguiente:

Programa de Procesamiento de Información del Módulo de Temperatura y Humedad Relativa	ADC.h	Obtención de datos digitales
	muestreo.h	Conversión ADC en intervalos de tiempo definidos
	UART.h	Transmisión asincrónica utilizando el interfaz UART
	LCD.h	Manejo de Comandos del LCD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2

**Figura 3.23 Programa de Procesamiento de Información dentro del Esquema del Conjunto de Programas del Módulo de Temperatura y Humedad Relativa**

*Fuente: Elaboración propia*

1) Definición de librerías para el manejo de:

- Entradas y salidas digitales (avr/io.h).
- Subrutinas de interrupciones (avr/interrupt.h).
- Operaciones matemáticas básicas (math.h).
- Operaciones con cadenas de caracteres (string.h).
- Variables tipo enteras (stdint.h).
- Introducción de retardos de tiempo (avr/delay.h).
- Conversiones de cadenas de caracteres y valores numéricos (stdlib.h).

<sup>7</sup> El programa principal se muestra en el Anexo A.1, desarrollado únicamente para el manejo del procesamiento de información del módulo de temperatura y humedad relativa.

- Operaciones estándar de entrada y salida o presentación de datos (stdio.h).
- Memoria EEPROM (avr/eeprom.h).

2) Declaración de librerías previamente estructuradas y definidas como:

- Conversión ADC simple (adc.h).
- Manejo del LCD (lcd.h).
- Muestreo de la conversión ADC (muestreo.h).
- Manejo del interfaz UART (uart.h).
- Manejo de interrupciones externas (interrupcion\_externa.h).

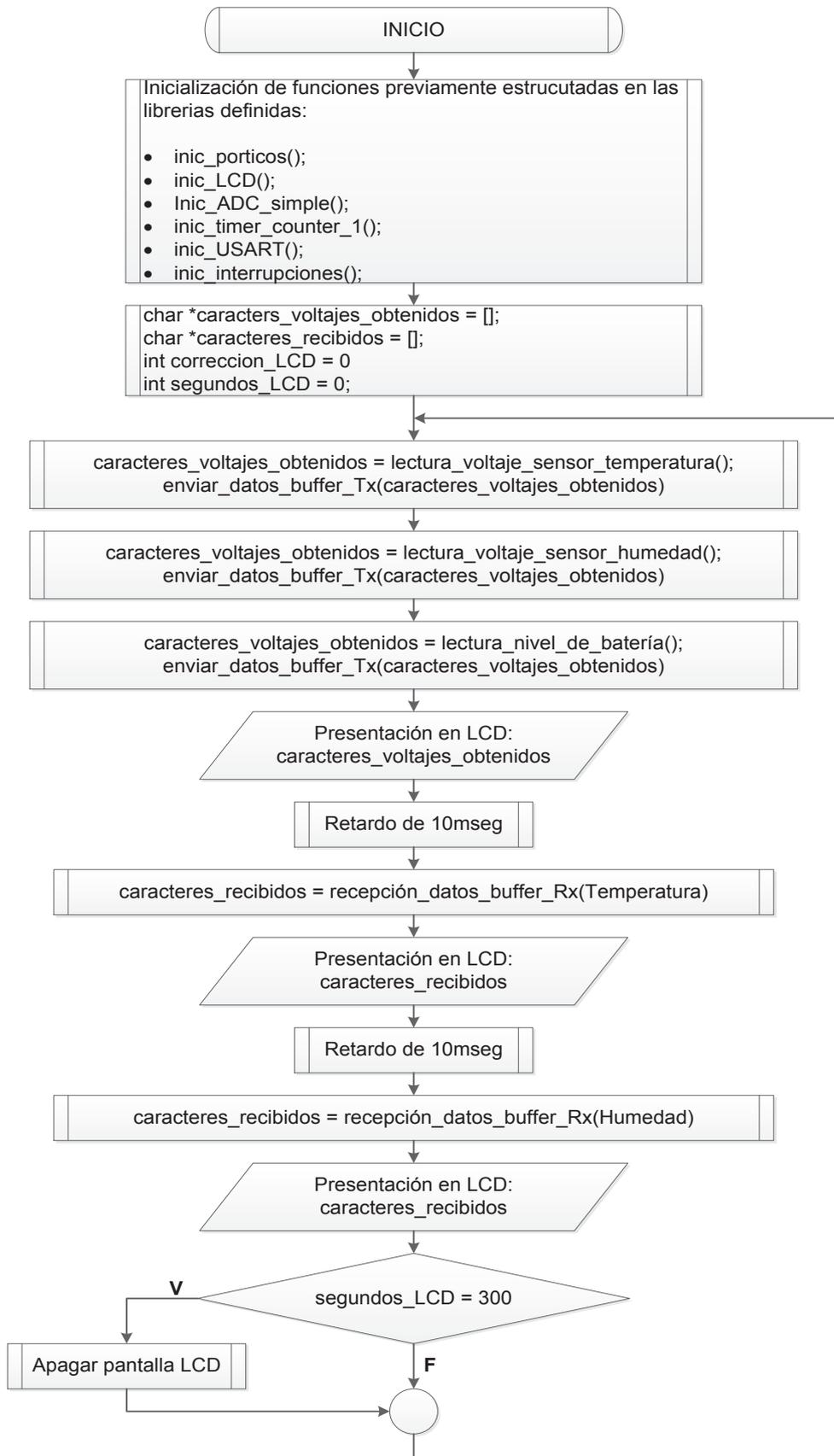
3) Declaración de variables globales, las cuales cumplen con las siguientes funciones:

- caracteres\_canal\_ADC\_obtenidos, es una variable tipo char, que permite obtener la cadena de caracteres de los valores de voltaje del sensor temperatura y humedad relativa.
- caracteres\_recibidos, es una variable tipo char, que permite obtener la cadena de caracteres de los valores de temperatura y humedad relativa procesados por la unidad de adquisición y procesamiento de datos meteorológicos.
- caracteres\_nivel\_batería\_obtenidos, es una variable tipo char, que permite obtener la cadena de caracteres referente al porcentaje de nivel de batería.

4) Definición de funciones, las cuales se enumeran como:

- inic\_porticos, para inicialización y configuración de los pódicos de entradas/salidas digitales de propósito general.
- inic\_interrupciones, para inicialización de las interrupciones de los subsistemas que se requieran utilizar del AVR.
- lectura\_voltaje\_sensor\_temperatura, para lectura del canal AD0 en el cual se encuentra conectado el sensor de temperatura.
- lectura\_voltaje\_sensor\_humedad, para lectura del canal AD2 en el cual se encuentra conectado el sensor de humedad.

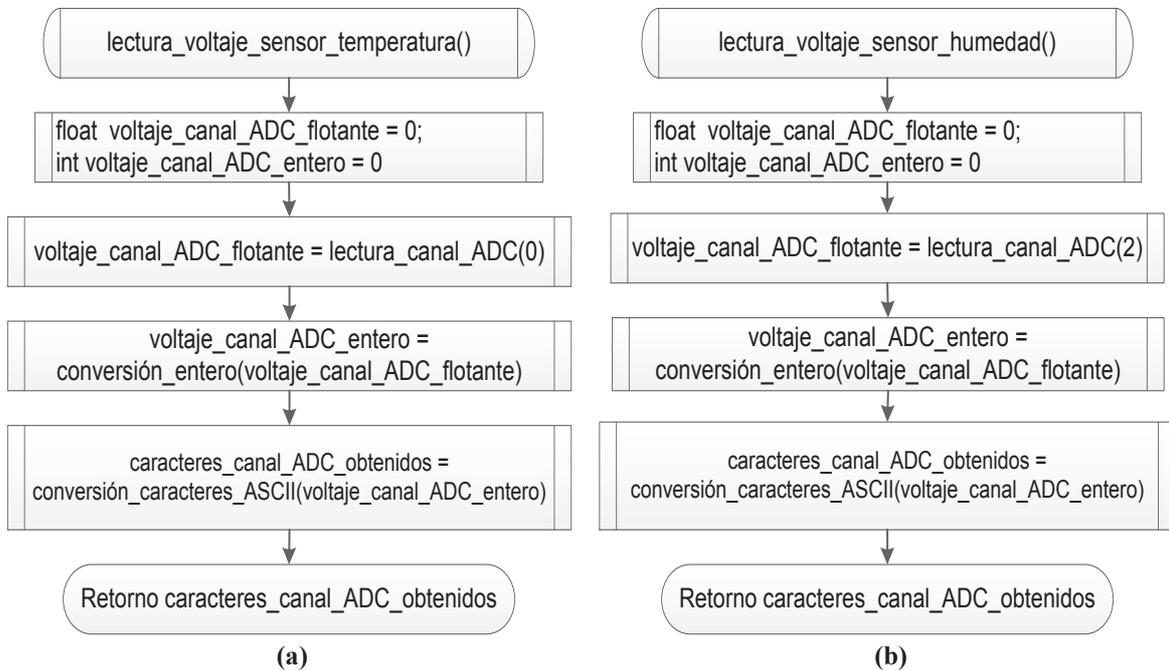
- lectura\_nivel\_bateria, para lectura del canal AD1 en el cual se encuentra conectado el detector de nivel de voltaje necesario para funcionamiento del circuito.
- 5) Función principal representada mediante el diagrama de flujo de la Figura 3.24.
- 6) Función de inicialización de pódicos para configuración de los diferentes pódicos A, B, C y D como entradas/salidas digitales de propósito general, para cumplir con las siguientes tareas:
- Salidas de datos al LCD mediante la utilización de los pines PC0-PC7.
  - Habilitación de los pines PA5-PA7 como salida de datos, para envío de comandos a los pines RS, RW y E respectivamente del LCD.
  - Habilitación de los canales de conversión ADC: AD0, AD1 y AD2, mediante la utilización de los pines PA0-PA2 como entrada de datos.
  - Habilitación de la recepción del interfaz USART0, mediante la utilización del pin PD0 como entrada de datos.
  - Habilitación de la transmisión del interfaz USART0, mediante la utilización de pin PD1 como salida de datos.
  - Habilitación de los pines PD2 y PD3 como entrada de datos, para detección de interrupciones externas.
  - Activación de los pull-ups del resto de pines que no se utilizan.
- 7) Función de inicialización de las interrupciones, para habilitación de:
- Interrupción ADC.
  - Interrupción de la detección de comparación de salida A y B del Timer/Counter1.
  - Interrupción de recepción completa del interfaz USART0.
  - Interrupción externa INT1, la cual se activa con la detección de una transición negativa.



**Figura 3.24 Diagrama de Flujo de la Función Principal del Programa del Módulo de Temperatura y Humedad Relativa**

*Fuente: Elaboración propia*

- 8) Función de lectura de voltaje del sensor de temperatura, descrito en el diagrama de flujo de la Figura 3.25(a).



**Figura 3.25 Función de Lectura del Voltaje: (a) Sensor de Temperatura y (b) Sensor de Humedad**

*Fuente: Elaboración propia*

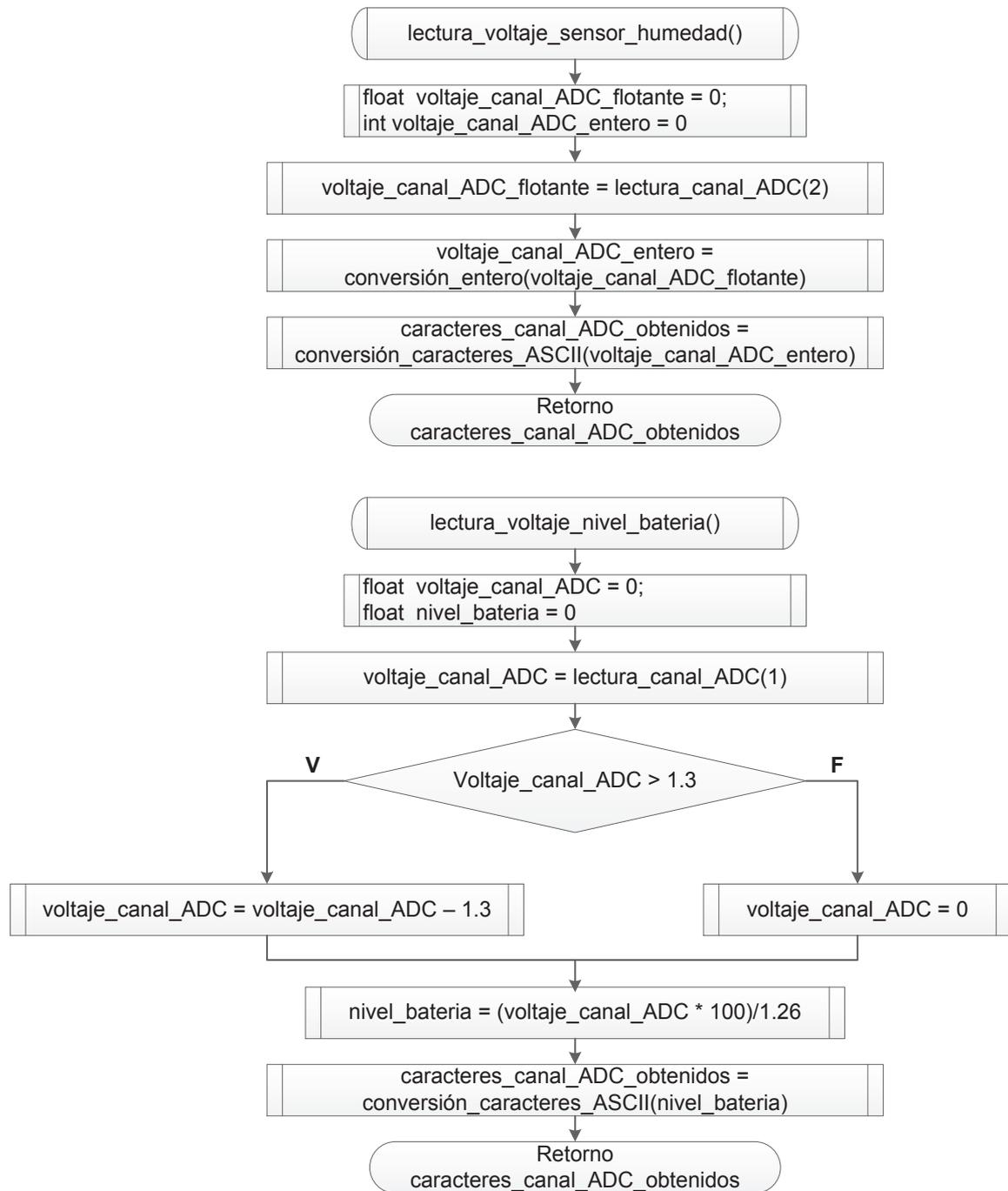
- 9) Función de lectura de voltaje del sensor de humedad, cuyo diagrama de flujo es el mismo que la función del punto 8, con la excepción de que en vez de leer el canal AD0, se procede a la lectura del canal AD2 tal como se muestra en la Figura 3.25(b).

- 10) Función de lectura del nivel de voltaje necesario para funcionamiento del circuito, descrito el diagrama de flujo de la Figura 3.26.

- 11) Fin del Programa.

### 3.2.2.7. Procedimiento para Configuración de un Dispositivo de Tx/Rx ZigBee

El procedimiento de configuración de un dispositivo de Tx/Rx ZigBee, se realiza tomando en cuenta las funciones que los dispositivos RFD y FFD desempeñan de acuerdo al Capítulo 1. De igual manera el procedimiento se da en base a los diferentes puntos mostrados en la Figura 3.3(b) tanto para dispositivos finales como coordinadores.



**Figura 3.26** Función de la Lectura del Nivel de Voltaje Necesario para Funcionamiento del Circuito.  
Fuente: Elaboración propia.

### 3.2.2.7.1. Configuración del Interfaz Serial UART

La configuración del interfaz serial UART se realiza de acuerdo a los siguientes pasos:

- a) Definición de la velocidad de transmisión del interfaz serial entre el módulo RF y el microprocesador. La velocidad de transmisión del interfaz RF no es

afectado por este parámetro, sin embargo, si la velocidad de transmisión de la interfaz serial es mayor que la velocidad de transmisión del interfaz RF, será necesario implementar un control de flujo. La Tabla 3.7 muestra las diferentes opciones de configuración de la velocidad de transmisión del interfaz serial.

**Tabla 3.7 Opciones de Configuración de la Velocidad de Transmisión Serial mediante el Comando ATBD.**

<b>Comando AT: ATBD</b>	
<b>Parámetro</b>	<b>Configuración (bps)</b>
0	1200
1	2400
2	4800
3	9600
4	19200
5	38400
6	57600
7	115200
Parámetro por defecto: 3	

*Fuente: [28]*

- b) Establecer el tipo de paridad a ser utilizado en el interfaz UART. Los tipos de paridad que pueden ser detectados son: par, impar, de marca, o de espacio, tal como se muestra en la Tabla 3.8. Se debe mencionar que el módulo no calculará, ni chequeará la paridad, solo establecerá el interfaz con dispositivos que si realicen el cálculo y chequeo de paridad.

**Tabla 3.8 Opciones de Configuración de Paridad del Interfaz UART mediante el comando ATNB**

<b>Comando AT: ATNB</b>	
<b>Parámetro</b>	<b>Configuración</b>
0	8-bit (no paridad) o 7-bit (cualquier paridad)
1	8-bit par
2	8-bit impar
3	8-bit marca
4	8-bit espacio

*Fuente: [28]*

- c) Configuración del tiempo de retardo requerido antes de realizar una transmisión. Este parámetro se utiliza para establecer el número de caracteres que se requiere almacenar en el buffer de Tx antes de realizar la transmisión. El número de caracteres a ser configurados pueden variar de 0 a FFhex y se realiza mediante el comando R0. En el caso en el que R0 = 0L, permitirá la transmisión de caracteres una vez que los mismos sean recibidos, en vez de almacenarlos en un paquete RF.

- d) Habilitar la operación API cambiando las tramas de modo transparente UART, por tramas API. La Tabla 3.9 muestra las diferentes opciones que permiten habilitar o des-habilitar la opción de tramas API.

**Tabla 3.9 Opciones de Configuración de la Habilitación de Operación en Modo Transparente o Operación mediante la utilización de Tramas API, por medio del Comando ATAP**

Comando AT: ATAP	
Parámetro	Configuración
0	Des-habilitado (operación transparente del interfaz UART)
1	Habilitación de la utilización de las tramas API
2	Habilitación de la utilización de las tramas API con caracteres de espacio

Fuente: [28]

#### 3.2.2.7.2. Configuración del Interfaz RF.

La configuración del interfaz RF, se realiza de acuerdo a los siguientes pasos:

- a) Determinación del nivel de potencia de transmisión del módulo. La Tabla 3.10 muestra las diferentes opciones de niveles de potencia de transmisión.

**Tabla 3.10 Opciones de Configuración para el Establecimiento del Nivel de Potencia de Trasmisión mediante el Comando ATPL**

Comando AT: ATPL	
Parámetro	Configuración
0	10dBm
1	12dBm
2	14dBm
3	16dBm
4	18dBm

Fuente: [28]

- b) Establecer un umbral sobre el cual se descarten los paquetes. Este parámetro da prioridad de transmisión de paquetes. El CCA (*Clear Channel Assessment*) es configurado para detectar el nivel de energía de una canal de transmisión. Si el nivel de energía detectado está por encima del umbral del CCA, el módulo RF no transmitirá el paquete.

#### 3.2.2.7.3. Configuración de la Asociación y Topología de Red de un Dispositivo Coordinador

La configuración de asociación y topología de red de un dispositivo FFD es:

- a) Especificar que el dispositivo es FFD o coordinador.
- b) Definir si el módulo coordinador realizará exploración activa o censado del canal de comunicaciones en busca de una dirección PAN ID disponible mediante el bit 0 del comando A2. En caso de que el módulo coordinador no realice un exploración activa, se debe establecer la dirección ID de la red PAN del módulo RF, tomando en cuenta que solo módulos con PAN IDs iguales pueden comunicarse entre sí. La configuración del parámetro en 0xFFFF indica una transmisión global para todas las PANs. No indica una recepción global.
- c) Determinar si el módulo coordinador realizará detección del nivel de energía para encontrar un canal de comunicaciones libre mediante la utilización del bit 1 del comando A2. En caso de no realizar detección del nivel de energía, se especificará el canal RF de operación. Para que los módulos se comuniquen entre sí, los módulos deben compartir el mismo canal. Diferentes canales pueden ser usados para prevenir que módulos de una red escuchen transmisiones de otra red. El módulo usa número de canales del estándar 802.15.4 de tal manera que:

$$\text{Frecuencia Central} = 2.405 + (\text{Canal} - 11) * 5\text{MHz} \quad \text{(Ecuación 3.12)}$$

Establecer el permiso de asociación al coordinador, para permitir la asociación de dispositivos finales al dispositivo coordinador mediante la activación del bit 2 del comando A2. En la Tabla 3.11 se observa las opciones de configuración del módulo coordinador mediante la utilización del comando ATA2.

#### 3.2.2.7.4. *Configuración de la Asociación y Topología de Red de un Dispositivo Final RFD*

La configuración de topología de red y de asociación de un dispositivo final RFD se describe en los siguientes pasos:

**Tabla 3.11 Opciones de Configuración de un Dispositivo Coordinador**

<b>Comando AT: ATA2</b>	
<b>Número de Bit</b>	<b>Opciones de Configuración del Dispositivo Coordinador</b>
0 – Asignación de la PAN ID	0L: El coordinador no realizará escaneo activo para localizar una identificación PAN ID disponible. En su lugar, operará con una PAN ID configurada previamente. 1L: El coordinador realizará escaneo activo para determinar un PAN ID disponible. Si se presenta un conflicto del identificador PAN ID, el parámetro ID será cambiado.
1 – Asignación del canal de comunicaciones	0L: El coordinador no realizará escaneo de nivel de energía para determinar un canal libre. En su lugar, se operará en un canal previamente configurado por el parámetro CH. 1L: El coordinador realizará escaneo de nivel de energía para encontrar un canal libre, para operar en el mismo.
2 – Habilitación de asociación	0L: El coordinador no permitirá la asociación de ningún dispositivo. 1L: El coordinador habilitará la asociación de dispositivos.
3-7	Reservado

*Fuente: [28]*

- a) Especificar que el dispositivo es RFD o final.
- b) Determinar el tipo de asociación con el módulo coordinador. El bit 0 del comando A1 permite determinar si se requiere realizar una asociación con el dispositivo coordinador que opera en una determinada PAN ID, o una asociación con un coordinador de cualquier PAN ID.
- c) Determinar el tipo de asociación con el canal de operación. Este parámetro de reasignación de canal configurado por medio del bit 1 del comando A1, permite la asociación con el coordinador que opera en un determinado canal o permite la asociación con un Coordinador de cualquier canal de operación.
- d) Establecer si se requiere que los intentos de asociación de un dispositivo final sea continua hasta tener éxito, por medio del bit 2 del comando A1.
- e) Especificar si se realizará una operación de poleo una vez que un determinado pin se active luego de estar en estado de reposo, para obtener cualquier dato

pendiente por parte del dispositivo coordinador, por medio del bit 3 del comando A1.

La Tabla 3.12 especifica la configuración de un dispositivo final por medio de las opciones que presenta el comando ATA1.

**Tabla 3.12 Opciones de Configuración de un Dispositivo Final.**

<b>Comando AT: ATA1</b>	
<b>Número de Bit</b>	<b>Opciones de Asociación del Dispositivo Final</b>
0 – Asignación de la PAN ID	0L: El dispositivo final solo se asociará con el dispositivo coordinador que se encuentre operando con una PAN ID previamente configurada. 1L: El dispositivo final se asociará con un dispositivo coordinador que se encuentre operando en cualquier PAN ID.
1 – Asignación del canal de comunicaciones	0L: El dispositivo final solo se asociará con un dispositivo coordinador que se encuentre operando en un canal de comunicación previamente configurado. 1L: El dispositivo final se asociará con un dispositivo coordinador que se encuentre operando en cualquier canal de comunicaciones.
2 – Habilitación de auto asociación	0L: El dispositivo no realizará una operación de asociación 1L: El dispositivo realizará una operación de asociación hasta tener éxito.
3 – Operación de poleo	0L: La activación de un pin luego del estar en estado de reposo no realizará una petición de datos pendientes por parte del dispositivo coordinador 1L: La activación de un pin luego del estar en estado de reposo, realizará una petición de datos pendientes por parte del dispositivo coordinador
4-7	Reservado

Fuente: [28]

#### 3.2.2.7.5. Configuración del Direccionamiento de Origen y Destino de cada Módulo

La configuración de direccionamiento de origen y destino, se da de acuerdo a la siguiente secuencia de pasos:

- a) Establecimiento del direccionamiento de origen mediante el comando MY, tomando en cuenta que la configuración de MY en 0xFFFF<sup>8</sup>, permite deshabilitar la recepción de paquetes RF con un direccionamiento de 16-bits, y

<sup>8</sup> Se debe considerar que el prefijo 0x..... indica un valor hexadecimal.

habilita el direccionamiento de 64-bits correspondiente al número serial del dispositivo.

- b) Lectura o escritura de los 32 bits más significativos de la dirección de destino de 64-bits del módulo RF, mediante la utilización del comando DH.
- c) Lectura o escritura de los 32 bits menos significativos de la dirección de destino de 64-bits del módulo RF, mediante la utilización del comando DL.

Un módulo solo se comunicará con otro módulo si presenta el mismo canal (parámetro CH), el mismo PAN ID (parámetro ID) y la misma dirección de destino (parámetros DH + DL).

Para transmitir usando una dirección de 16-bits, se debe configurar el parámetro DH en cero y el parámetro DL menor que 0xFFFF. La dirección 0x00000000000000FFFF (DL contenido en DH) es de broadcast para la red PAN.

#### 3.2.2.7.6. *Configuración del Método de Acceso al Medio MAC*

La configuración del método de acceso al medio MAC, se realiza tomando en cuenta los siguientes pasos:

- a) Establecer el valor del exponente del algoritmo de Back-off de CSMA/CA. El algoritmo CSMA-CA es implementado para evitar colisiones (retardos aleatorios son implementados para prevenir pérdida de datos causadas por colisiones).
- b) Establecer el número máximo de reintentos de re-transmisión. Este parámetro puede ser instituido mediante el comando RR. El número de reintentos que el dispositivo efectúa por defecto es de 3, de acuerdo al estándar IEEE 802.15.4.

#### 3.2.2.8. **Parámetros y Valores de Configuración del Dispositivo de Tx/Rx ZigBee RFD del Módulo de Temperatura y Humedad Relativa**

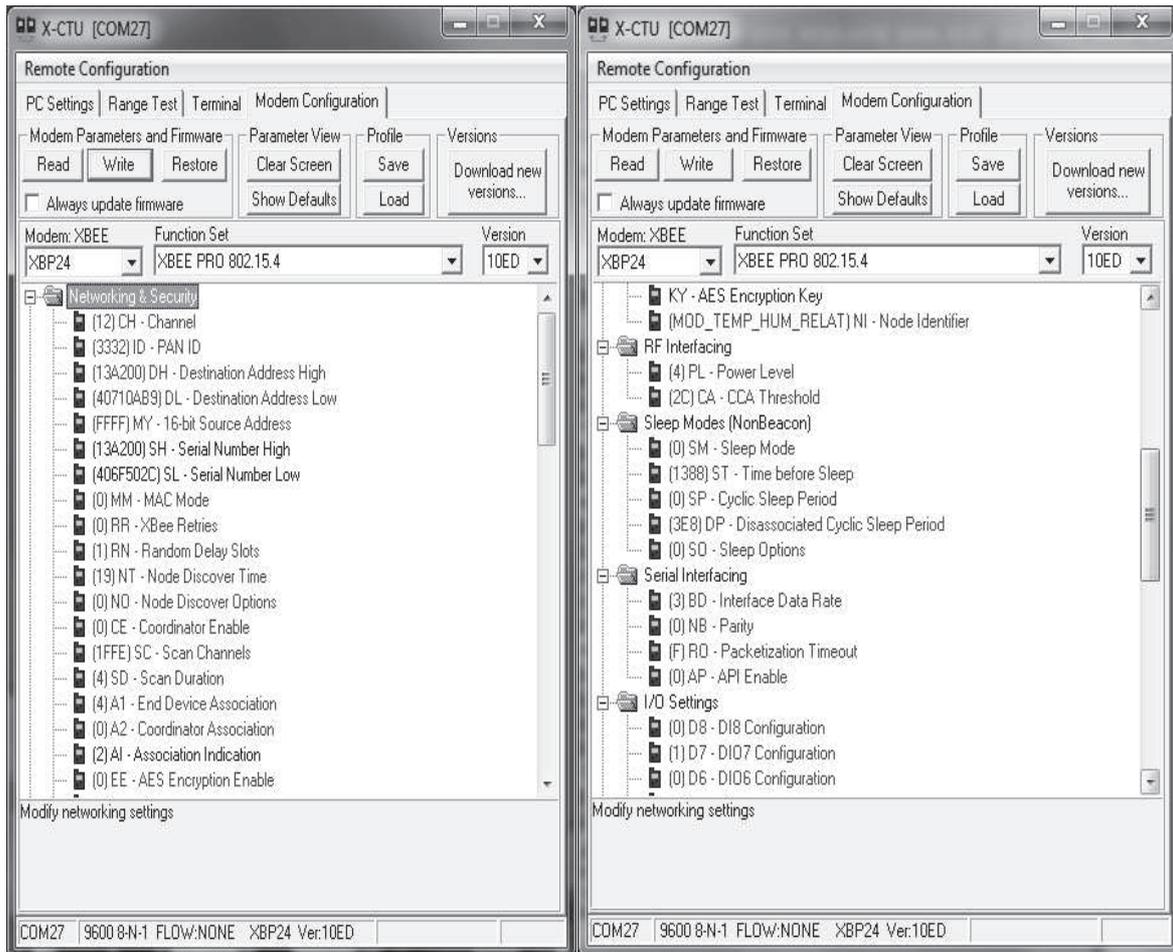
Tomando en cuenta el procedimiento de configuración de un dispositivo ZigBee descrito en la sección anterior, se presenta en la Tabla 3.13 los valores y

parámetros que se configurarán para el caso del dispositivo de Tx/Rx ZigBee RFD del módulo de temperatura y humedad relativa. Mientras tanto, la Figura 3.27, muestra los parámetros configurados utilizando el programa X-CTU.

**Tabla 3.13 Parámetros y Valores de Configuración del Dispositivo RFD ZigBee del Módulo de Temperatura y Humedad Relativa**

Configuración	Parámetros	Comando	Valor del Parámetro	Descripción
Interfaz Serial UART	Velocidad de transmisión	ATBD	3	9600bps
	Paridad	ATNB	0	No-paridad
	Tiempo de retardo	ATRO	0x0F	Número de caracteres almacenados en el buffer antes de ser enviados
	Operación API	ATAP	0	Des-habilitación de la recepción de tramas API del módulo coordinador
Interfaz RF	Nivel de Potencia	ATPL	4	Potencia de transmisión de 18dBm
	Umbral CCA	ATCA	0x2C	Umbral de -44dBm sobre el cual se descartan paquetes de recepción
Topología y Asociación	Tipo de Dispositivo	ATCE	0	Dispositivo final
	Tipo de asociación con el módulo coordinador	ATA1	Bit-0 = 0	Asociación con un dispositivo coordinador cuyo PAN ID es determinado por el identificador del nodo
	Canal de operación		Bit-1 = 1	Asociación con un dispositivo coordinador que opera en cualquier canal disponible
	Auto-asociación		Bit-2 = 1	El dispositivo intentará asociarse hasta tener éxito
	Operación de poleo		Bit-3 = 0	No se realizará operación de poleo luego de la activación del pin
Dirección de origen	ATMY		0xFFFF	Utilización del serial del módulo como dirección de origen: SL = 0x406F502C, SH = 0x13A200
Direccionamiento	Dirección de destino (32 bits más significativos)	ATDH	0x13A200	El módulo coordinador solo utilizará 32 bits menos significativos de direccionamiento de destino
	Dirección de destino (32 bits menos significativos)	ATDL	0x40710AB9	Dirección de destino del módulo coordinador
	Dirección ID de la red PAN	ATID	0x3332	Igual dirección que el módulo coordinador
Método de Acceso al Medio CSMA-CA	Valor del exponente de Back-off	ATRN	1	Valor mínimo del exponente de Back-off = (valor entero entre 0 y $2^{BE} - 1$ )x (aUnitBackoffPeriod)
	Número máximo de reintentos	ATRR	0	Se ejecutará solo 3 reintentos especificados por el estándar IEEE 802.15.4

*Fuente: Elaboración propia basada en la sección 3.2.2.7.*



**Figura 3.27** Ventanas del Software X-CTU que muestran los Parámetros de Configuración del Módulo RFD ZigBee de Temperatura y Humedad Relativa

Fuente: X-CTU.

### 3.2.3. DESCRIPCIÓN Y DISEÑO DEL SISTEMA DE ENERGIZACIÓN DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA

El sistema de energización es un sistema fotovoltaico, cuya función es la de producir energía eléctrica, a partir de la insolación. El sistema consta de los siguientes elementos [29]:

- **Un generador solar**, compuesto por un conjunto de paneles fotovoltaicos, que captan la insolación luminosa procedente del sol y la transforman en corriente continua de baja tensión (12 V o 24 V), tal como se muestra en la Figura 3.28.
- **Un acumulador o batería**, que almacena la energía producida por el generador y permite disponer de corriente eléctrica fuera de horas de luz o días nublados. La batería utilizada para este sistema se muestra en la Figura 3.29.

- **Un regulador de carga**, cuya misión es evitar sobrecargas o descargas excesivas al acumulador, tal como se muestra en la Figura 3.30.

### 3.2.3.1. Determinación y Características de los Elementos del Sistema Fotovoltaico

Los elementos del sistema fotovoltaico serán seleccionados de acuerdo a los equipos y dispositivos disponibles en el INAMHI, los cuales se presentan a continuación:

#### Panel Solar NAPS de la Estación Vaisala.

Tipo de Módulo:	NP25R
Potencia Máxima (+/-10%):	25 Wp
Voltaje Nominal (Vn):	12 V
Corriente de Operación Imp Típico:	1.44 A
Voltaje Operativo Vmp Típico:	17.4 V
Corriente de Cortocircuito (Isc):	1.60 A
Voltaje de Circuito Abierto Voc típico:	21.6 V



**Figura 3.28 Panel Solar de 12 V con un ISC de 1.6 A**

*Fuente: [30]*

#### Batería CSB.

Voltaje Nominal:	12 V
Capacidad:	7 Ah
Número de días de autonomía del sistema fotovoltaico:	1 día.
Profundidad de descarga:	80%



**Figura 3.29** Batería CSB de 12 V, 7 Ah

Fuente: [31]

### **Regulador o Controlador de Carga Solar SunSaver-10.**

Voltaje Nominal:	12 V
Entrada Nominal Solar:	6.5 A
Carga Nominal:	10 A



**Figura 3.30** Controlador de Carga Solar Sun Saver-6

Fuente: [32]

#### **3.2.3.2. Diseño del Sistema Fotovoltaico [29]**

El procedimiento para dimensionamiento del sistema fotovoltaico, se realizó de acuerdo a las recomendaciones de diseño mostradas en el documento Atlas Solar del Ecuador con Fines de Generación Eléctrica, presentado por el Conelec. Por tanto se presenta los siguientes pasos:

- 1) Determinación de las cargas en corriente continua.
- 2) Especificación de la corriente pico del módulo.
- 3) Dimensionamiento del módulo fotovoltaico.
- 4) Dimensionamiento del banco de baterías.

A continuación se describen de forma detallada los pasos de diseño mencionados previamente.

### 3.2.3.2.1. Determinación de las Cargas en Corriente Continua (12 VCC)

La determinación de cargas se da en base al consumo de energía de los dispositivos que conforman el módulo de temperatura y humedad relativa.

**Tabla 3.14 Consumo de Energía de los Dispositivos Principales del Módulo de Temperatura y Humedad Relativa**

Equipo/ Dispositivo	Consumo de Voltaje (V) A1	Consumo de Corriente (mA) A2	Potencia (W) A3=A1xA2	Cantidad A4	Total(W) A5=A3xA4	Número de horas de uso por día	Consumo de energía (Wh) por día A7=A5xA6
XBee-PRO 802.15.4 OEM RF Modules	3.4	340	1.156	1	1.156	24	27.744
Microcontrolador Atmega164P	5	200	1	1	1	24	24
Sensor HMP45A/D	12	4	0.048	1	0.048	24	1.152
Carga CC total diaria:							52.896

*Fuente: Elaboración propia basada en las hojas de características de los dispositivos mencionados en la tabla.*

A8: Carga diaria CC equivalente = 52.896 Wh/día.

### 3.2.3.2.2. Especificación de la Corriente Pico del Módulo

En este paso se pretende encontrar la corriente pico de consumo del sistema:

B1: Carga diaria CC = A8 = 52.896 Wh/día.

B2: Tensión CC del Sistema = 12 V

B3: Carga diaria de corriente CC = B1/B2 = 4.408 Ah

B4: Factor de seguridad (pérdidas del sistema) = 1.2

B5: Carga de corriente corregida = B3 x B4 = 5.29 Ah

B6: Valor insolación global promedio = 4.575 KWh/m<sup>2</sup>

B7: Corriente pico del sistema: B5/B6 = 1.16 A

### 3.2.3.2.3. Dimensionamiento del Módulo Fotovoltaico.

En esta sección se determina el número de módulos fotovoltaicos necesarios para cubrir con la corriente pico del sistema.

C1: Corriente de cortocircuito I<sub>sc</sub> del panel solar = 1.6 V

C2: Arreglo de módulos en paralelo = B7/C1 = 0.722 ≈ 1

C3: Tensión CC nominal del sistema: 12 V

C4: Tensión CC nominal del módulo: 12 V

C5: Número total de módulos:  $(C3/C4)*C2 = 1$

#### 3.2.3.2.4. Dimensionamiento del Banco de Baterías.

En esta sección se procede a determinar el número de baterías necesarias para almacenamiento de energía producida por el generador y disposición de corriente eléctrica fuera de horas de luz.

D1: Carga CC total diaria = B5 = 5.29 Ah.

D2: Número de días de autonomía del sistema fotovoltaico = 1 día.

D3: Capacidad nominal banco de baterías = D1 x D2 = 5.29 Ah.

D4: Profundidad de descarga de la batería CSB = 80%.

D5: Capacidad corregida banco de baterías:  $D3/D4 = 6.61$  Ah.

D6: Capacidad nominal de la batería CSB: 7 Ah.

D7: Arreglo de baterías en paralelo:  $D5/D6 = 1$

D8: Tensión CC nominal del sistema = 12 V.

D9: Tensión CC nominal de la batería CSB = 12 V.

D10: Número total de baterías =  $(D8/D9) \times D7 = 1$ .

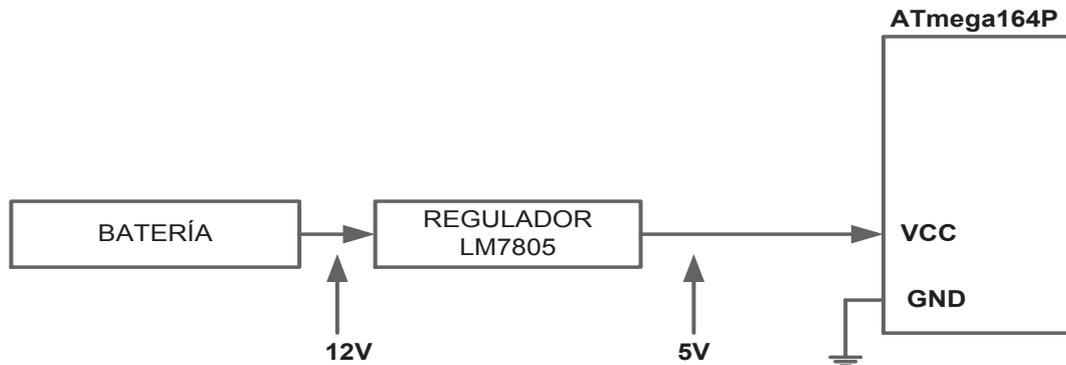
En resumen es necesario un solo módulo fotovoltaico, una batería y un solo controlador de carga, cuyas características técnicas fueron descritos en la sección 3.2.3.1.

#### 3.2.4. DESCRIPCIÓN Y DISEÑO DEL SISTEMA DE REGULACIÓN Y DETECCIÓN DE VOLTAJE DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA

El sistema de regulación y detección de nivel de voltaje está conformado por un circuito de regulación de 5V para funcionamiento del microprocesador y periféricos, un circuito de regulación de 3.5V para funcionamiento del dispositivo de Tx/Rx ZigBee RFD y un circuito de detección de nivel de voltaje obtenido de los pines LOAD del controlador de carga.

### 3.2.4.1. Circuito de Regulación de 5 V

El circuito de regulación de 5 V, está conformado por el circuito integrado LM7805 alimentado por una batería de 12 V, tal como se observa en la Figura 3.31.



**Figura 3.31** Diagrama de Bloques del Circuito de Regulación de 5V que Alimenta un Microcontrolador Atmega164P

*Fuente: Elaboración propia*

El circuito integrado LM7805 presentado en la Figura 3.32, presenta las siguientes características:

$I_O = 5 \text{ mA to } 1 \text{ A}$ ,

$C_O = 0.1 \text{ uF}$

$P_O \leq 15 \text{ W}$ ,

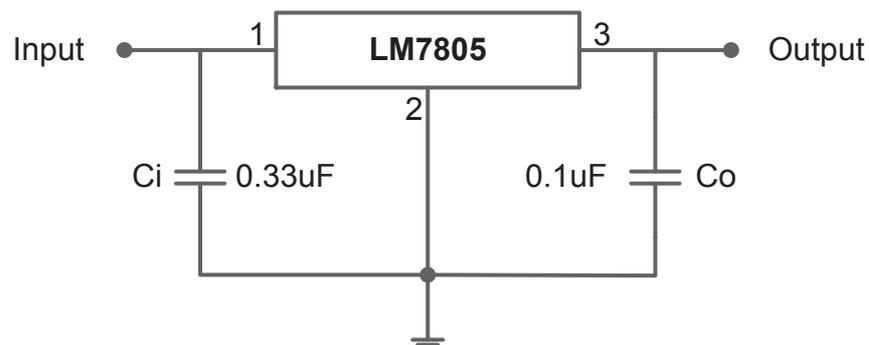
$V_{\text{omin}} = 4.8 \text{ V}$

$V_I = 6 \text{ V to } 20 \text{ V}$

$V_{\text{otyp}} = 5 \text{ V}$

$C_I = 0.33 \text{ uF}$

$V_{\text{omax}} = 5.2 \text{ V}$



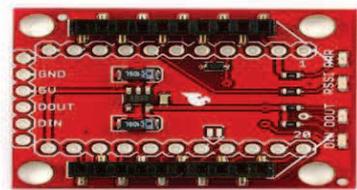
**Figura 3.32** Diagrama Circuito Integrado LM7805 para Regulación de Voltaje de 5 V

*Fuente: [33]*

### 3.2.4.2. Circuito de Regulación de 3.3 V

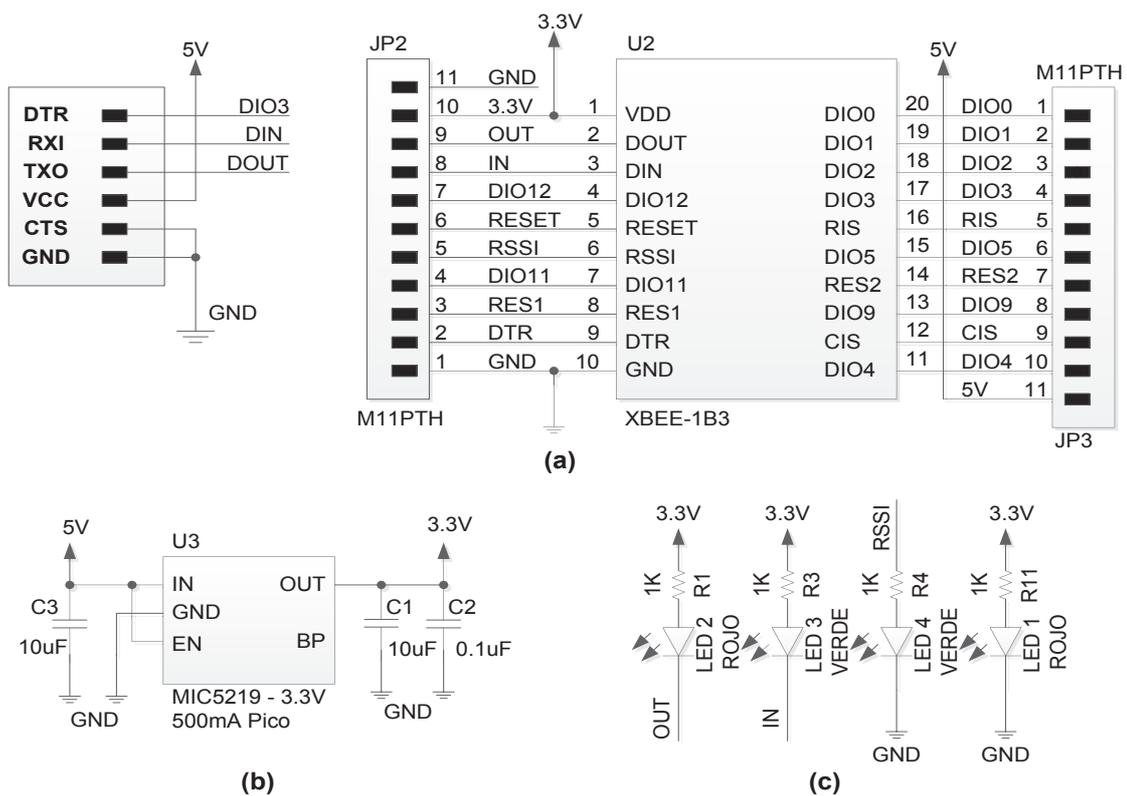
La regulación de 3.3V se realiza por medio del circuito integrado XBee Regulator del fabricante SparkFun, mostrado en la Figura 3.33. Este circuito a más de la

regulación, se encarga del acondicionamiento de las señales e indicadores de actividad (encendido, RSSI y DIN/DOOUT). Permite obtener señales de 3.3V a partir de 5V de polarización.



**Figura 3.33 XBee-Regulator de SparkFun.**  
Fuente: [34]

Los esquemas de conexiones del XBee Regulator con un módulo Tx/Rx XBee-PRO se presentan en la Figura 3.34.

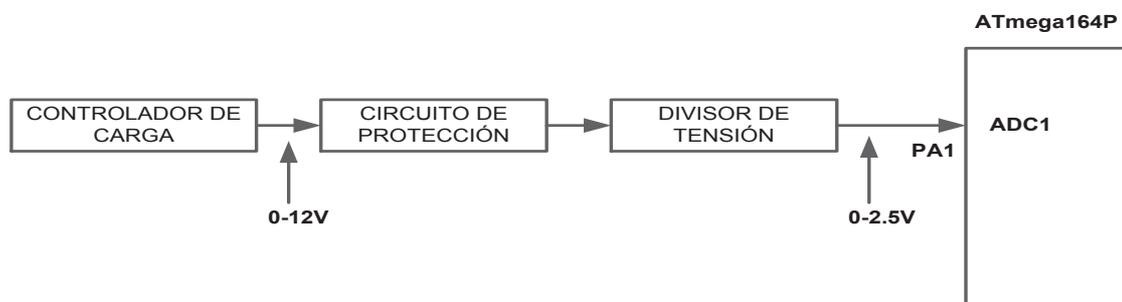


**Figura 3.34 Esquemas de: (a) Circuito de Conexiones del Dispositivo XBEE, (b) Circuito Integrado Regulador de 3.5 V y (c) Circuitos Leds de Indicación**  
Fuente: [34]

### 3.2.4.3. Circuito de Detección de Nivel de Voltaje

Este circuito cuyo diagrama de bloque se muestra en la Figura 3.35, tiene como propósito detectar el nivel de voltaje de carga del controlador solar de carga, para

indicar si el módulo de temperatura y humedad relativa tiene la energía necesaria de funcionamiento. Este circuito se encuentra conformado por un circuito de protección y un divisor de tensión que va permitir convertir valores de voltaje utilizado por el controlador de carga (0 - 12V), a valores de voltaje en los cuales los pines del microprocesador ATmega164P trabaja (0 - 2.5V). Los valores de voltaje van a ser ingresados en el pin PA1 correspondiente al canal de conversión ADC1, para su correspondiente procesamiento ya descrito anteriormente en la sección 3.2.2.1.



**Figura 3.35 Diagrama de Bloques del Circuito de Detección de Voltaje**  
 Fuente: Elaboración propia

#### 3.2.4.3.1. Circuito de Protección [35]

El esquema de protección tiene como propósito limitar la corriente del elemento o circuito en serie, que en este caso sería el circuito de detección de voltaje, y se diseña para estar inactivo bajo condiciones de operación normal y activarse tan pronto como se intente exceder el correspondiente límite de seguridad. Un circuito de protección de sobrecarga evita que la corriente que circula por el transistor en serie exceda un nivel de seguridad predeterminado, como sucedería por ejemplo en el caso de cortocircuitar la salida.

Se implementa entonces un circuito de protección denominado limitación de corriente foldback basado en el circuito descrito en la Figura 3.36(a). La idea de la limitación foldback es reducir la corriente y la tensión de la carga cuando ocurren las condiciones de sobrecarga. En la Figura 3.36(b) se muestra que cuando se alcanza la carga máxima IFB, el voltaje de salida disminuye y la corriente de la carga se reduce. Esto reduce la disipación de potencia del regulador y del resto de los componentes de la fuente de alimentación.

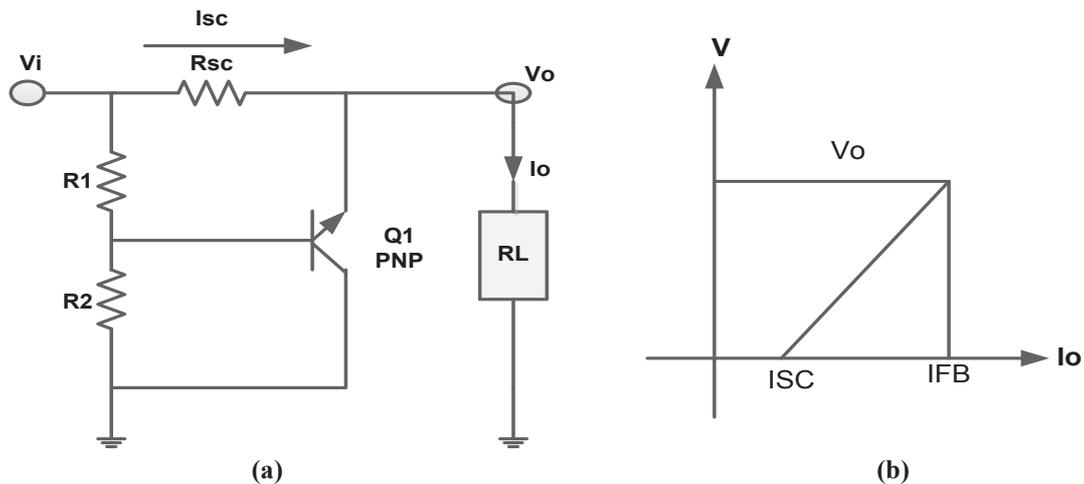


Figura 3.36 Esquemas de: (a) Circuito de Protección de Corriente "Foldback" y (b) Curva de Limitación de Corriente

Fuente: [35]

Bajo condiciones normales de carga  $R_1$ ,  $R_2$  y  $Q_1$  no tienen efecto en la operación del circuito. Bajo condiciones de sobrecarga, la caída de tensión en  $R_{SC}$  hará que  $Q_1$  conduzca, produciendo una disminución en la tensión de salida. La disminución de  $V_o$  reduce aún más la caída de tensión en  $R_1$ , haciendo que  $Q_1$  se sature y reduciendo aún más la corriente y la tensión de carga. Las intensidades de corriente  $I_{FB}$  e  $I_{SC}$  están dadas por las siguientes ecuaciones:

$$\text{Si } V_o = 0 \Rightarrow I_o = I_{SC} \text{ y } V_{BE1} = I_{SC} R_{SC} \frac{R_2}{R_1 + R_2} \quad (\text{Ecuación 3.13})$$

$$\Rightarrow I_{FB} = V_o \frac{R_1}{R_2 \times R_{SC}} + V_{BE} \frac{R_1 + R_2}{R_2 \times R_{SC}} \quad (\text{Ecuación 3.14})$$

$$\text{Si } I_L = I_{FB} \Rightarrow R_{SC} I_{FB} = (V_o + R_{SC} I_{FB}) \frac{R_1}{R_1 + R_2} + V_{BE} \quad (\text{Ecuación 3.15})$$

$$\Rightarrow I_{SC} = \frac{V_{BE}}{R_{SC}} \times \left( \frac{R_1 + R_2}{R_2} \right) \quad (\text{Ecuación 3.16})$$

La corriente de cortocircuito no debe ser muy pequeña; una buena elección es:

$$I_{SC} = \frac{I_{FB}}{3} \quad (\text{Ecuación 3.17})$$

A continuación se describe el proceso para determinar el  $R_{SC}$ :

- Despejando la (Ecuación 3.14):

$$I_{FB} = V_o \frac{R1}{R2R_{SC}} + V_{BE} \frac{R1 + R2}{R2 \times R_{SC}}$$

$$I_{FB} = \frac{V_o R1 + V_{BE} R1 + V_{BE} R2}{R2 R_{SC}}$$

$$R2(I_{FB} R_{SC} - V_{BE}) = R1(V_o + V_{BE}) \quad \text{(Ecuación 3.18)}$$

- Despejando la (Ecuación 3.16):

$$I_{SC} = \frac{V_{BE}}{R_{SC}} \times \left( \frac{R1 + R2}{R2} \right)$$

$$I_{SC}(R_{SC} R2) = V_{BE} R1 + V_{BE} R2$$

$$R2(I_{SC} R_{SC} - V_{BE}) = V_{BE} R1$$

$$R2 = \frac{V_{BE} R1}{I_{SC} R_{SC} - V_{BE}} \quad \text{(Ecuación 3.19)}$$

- Remplazando (Ecuación 3.18 en la (Ecuación 3.19):

$$\frac{V_{BE} R1}{I_{SC} R_{SC} - V_{BE}} (I_{FB} R_{SC} - V_{BE}) = R1(V_o + V_{BE})$$

$$V_{BE} I_{FB} R_{SC} - V_{BE}^2 = (V_o + V_{BE})(I_{SC} R_{SC} - V_{BE}) \quad \text{(Ecuación 3.20)}$$

- Si:  $I_{FB} = 3I_{SC}$

$$(V_{BE} \times 3I_{SC} R_{SC}) - V_{BE}^2 = I_{SC} R_{SC} V_o + I_{SC} R_{SC} V_{BE} - V_o V_{BE} - V_{BE}^2$$

$$R_{SC}(V_{BE} \times 3I_{SC} - I_{SC} V_o - I_{SC} V_{BE}) = -V_o V_{BE} \times (-1)$$

$$R_{SC}(I_{SC}V_O + I_{SC}V_{BE} - 3I_{SC}V_{BE}) = V_OV_{BE}$$

$$R_{SC} = \frac{V_OV_{BE}}{I_{SC}(V_O - 2V_{BE})} \quad \text{(Ecuación 3.21)}$$

Para determinar los correspondientes valores de resistencias del circuito de protección de la Figura 3.36, se debe considerar que la corriente de corto circuito  $I_{SC}$  es la corriente DC de los pines de entrada/salida I/O, cuyo valor es de 40 mA, tal como se muestra en la Tabla 3.15. El valor del voltaje de salida  $V_O$  es de 12V correspondiente a la batería de alimentación DC.

Se procede entonces a determinar los valores de los elementos del circuito de protección, tomando en cuenta la (Ecuación 3.21) y la (Ecuación 3.19).

**Tabla 3.15 Características y Valores Máximos de Operación Eléctricas del Microcontrolador ATmeg164P**

Parámetro	Descripción
Temperatura de operación	-55oC – +125oC
Voltaje en cada pin excepto en RESET con respecto a tierra	-0.5V – Vcc+0.5V
Voltaje en RESET con respecto a tierra	-0.5V – +13.0V
Voltaje máximo de operación	6.0V
Corriente DC en cada pin I/O	40.0mA
Corriente DC en Vcc y en los pines GND	200.0mA

Fuente: [24]

### Datos:

$$I_{SC} = 40\text{mA} \quad R_{sc} = ?$$

$$V_O = 12\text{V} \quad R_1 = ?$$

$$V_{BE} = 0.7\text{V} \quad R_2 = ?$$

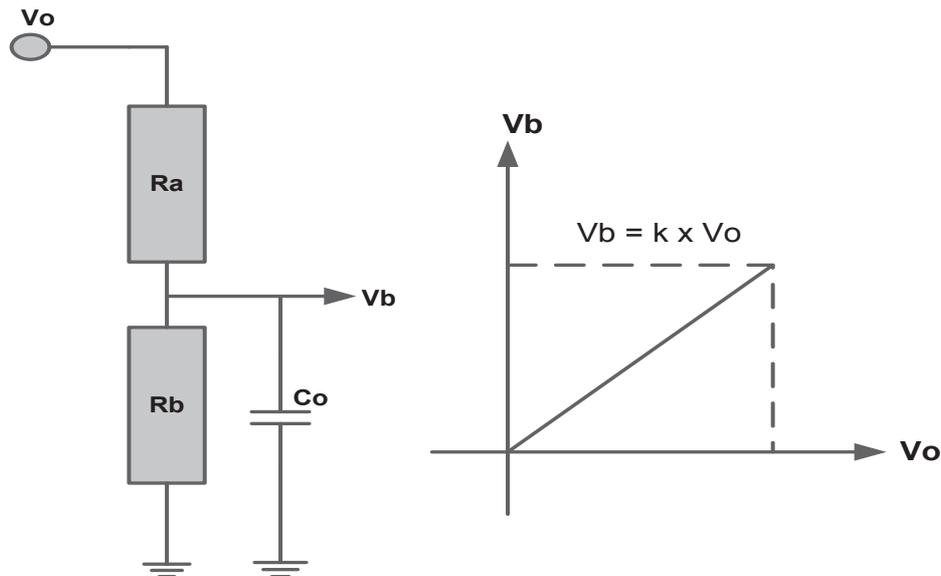
$$R_{SC} = \frac{V_OV_{BE}}{I_{SC}(V_O - 2V_{BE})} = \frac{12\text{V} \times 0.7\text{V}}{40\text{mA}(12\text{V} - 2 \times 0.7\text{V})} = 19.81\Omega \approx 20\Omega$$

$$R_2 = \frac{V_{BE}R_1}{I_{SC}R_{SC} - V_{BE}} = \frac{0.7\text{V} \times R_1}{(40\text{mA} \times 20\Omega - 0.7\text{V})} = 7R_1$$

$$\text{Si: } R_1 = 12\text{K}\Omega \rightarrow R_2 = 7 \times R_1 = 84\text{K}\Omega \approx 82\text{K}\Omega$$

### 3.2.4.3.2. Divisor de Tensión

Este circuito mostrado Figura 3.37, permite censar valores de voltaje de 0 - 12V, por medio de la obtención de voltajes de 0 – 2.56V manejados por el microprocesador ATmega164P. Se debe señalar que un circuito divisor de tensión permite relacionar voltajes de entrada y salida de forma lineal.



**Figura 3.37** Circuito Divisor de Tensión.

*Fuente: Elaboración propia.*

La ecuación que relaciona ambos voltajes es:

$$Vb = \frac{Rb}{Ra + Rb} Vo \quad \text{(Ecuación 3.22)}$$

Para determinar los valores de las resistencias, se debe despejar de la (Ecuación 3.22) cualquiera de ellas, y encontrar la relación correspondiente.

$$Vb(Ra + Rb) = RbVo$$

$$RaVb + RbVb = RbVo$$

$$Ra = \frac{Vo - Vb}{Vb} Rb$$

$$V_b(R_a + R_b) = R_b V_o$$

$$R_a V_b + R_b V_b = R_b V_o$$

$$R_a = \frac{V_o - V_b}{V_b} R_b \quad \text{(Ecuación 3.23)}$$

**Datos:**

$V_b = 2.56V$  (voltaje máximo que puede soportar una entrada del ATmega164P)

$V_o = 12V$  (voltaje máximo que entrega la batería)

$R_a = ?$

$R_b = ?$

$$R_a = \frac{V_o - V_b}{V_b} R_b = \frac{12V - 2.56V}{2.56V} R_b = 3.69 R_b$$

Si:  $R_b = 10 K\Omega \mapsto R_a = 3.69 R_b = 3.69 \times 10 K\Omega = 36.9 K\Omega \approx 36 K\Omega$

$$Z_o = X_{co} \parallel R_b = X_{co} \parallel 36 k\Omega = X_{co} \quad \text{(Ecuación 3.24)}$$

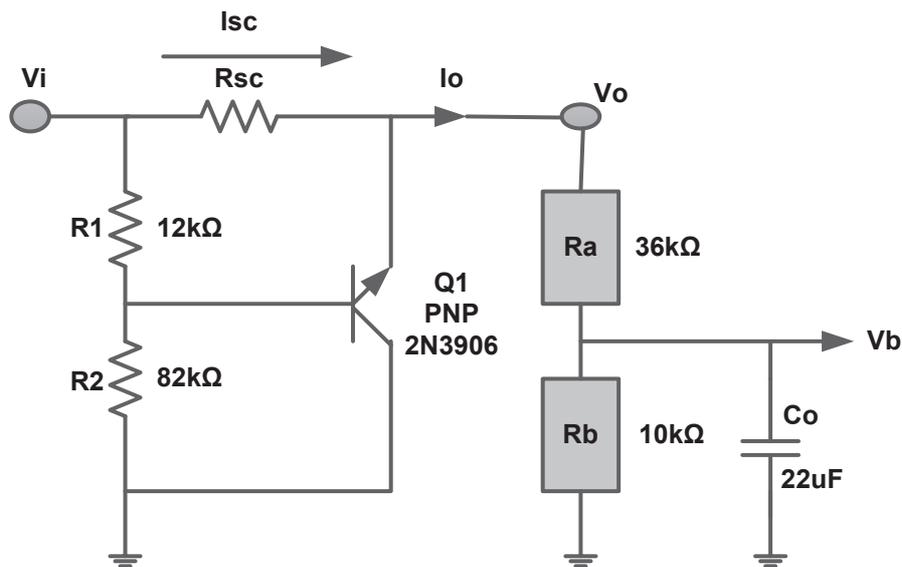
El capacitor  $C_o$  actúa como filtro para reducir interferencias de ruido en la entrada del canal ADC, siempre y cuando el valor de  $X_{co}$  sea mucho menor que  $R_b$  previamente calculado. Por lo tanto, se asume  $X_{co} = 8$ , valor que es mucho menor que  $R_b = 36 k\Omega$  y que a su vez permite obtener un valor de capacitancia estándar.

$$X_{co} = \frac{1}{\omega C_o} = \frac{1}{2\pi f C_o} \quad \text{(Ecuación 3.25)}$$

$$C_o = \frac{1}{2\pi f X_{co}} = \frac{1}{2\pi \times 1k \times 8} = 19.9 \mu F$$

$$C_o \approx 22 \mu F$$

La Figura 3.38 muestra el diseño del circuito de protección y del circuito de detección de voltaje con sus correspondientes valores de resistencias y capacitor.



**Figura 3.38** Circuitos de Protección y Detección de Voltaje.

Fuente: Elaboración propia.

### 3.2.4.3.3. Ecuación para Detección de Nivel de Voltaje en Porcentaje.

Si bien es cierto que el divisor de tensión permite representar el rango de 0 – 12V del voltaje de batería en valores de 0 – 2.56V manejados por los canales ADC del microprocesador, se debe también tomar bajo consideración que el voltaje mínimo para que el regulador LM7805 alimente al módulo de temperatura y humedad relativa es de 6V de acuerdo a las características mostradas en la sección 3.2.4.1. Por ende se procede a determinar el correspondiente valor que presenta el divisor de tensión para este voltaje, tomando en cuenta la (Ecuación 3.22 del divisor de tensión mostrada en la sección anterior.

#### **Datos:**

$$R_a = 36k\Omega$$

$$V_o = 5V$$

$$R_b = 10k\Omega$$

$$V_b = ?$$

$$V_b = \frac{R_b}{R_a + R_b} V_o = \frac{10k\Omega}{36k\Omega + 10k\Omega} \times 6V = 1.3V$$

De esta manera se observa un nuevo rango de operación de 6 – 12V, correspondiente a valores de 1.3V – 2.56V. Esto significa que el porcentaje que representa el funcionamiento del circuito debe estar dado en estos rangos. Por lo

tanto se procede a restar  $2.56V - 1.3V = 1.26V$ , cuyo resultado representa el 100% del nivel de voltaje de funcionamiento del circuito.

**Ejemplo:**

Si el voltaje de entrada  $V_{in}$  es de 8V, y no existe ningún tipo de sobre carga, entonces  $V_{in} = V_o = 8V$  de acuerdo al circuito mostrado en la Figura 3.38. Tomando en cuenta lo anteriormente expuesto, entonces el valor de voltaje del divisor de tensión se calcula como:

**Datos:**

$$R_a = 36k\Omega$$

$$V_o = 8V$$

$$R_b = 10k\Omega$$

$$V_b = ?$$

$$V_b = \frac{R_b}{R_a + R_b} V_o = \frac{10k\Omega}{36k\Omega + 10k\Omega} \times 8V = 1.74V$$

A continuación, se procede a determinar el valor de porcentaje tomando en cuenta que 1.3V es el valor mínimo del rango de voltaje previamente definido.

$$1.74V - 1.3V = 0.44V$$

$$1.26V \text{ ----- } 100\%$$

$$0.44V \text{ ----- } x = \frac{0.44}{1.26} \times 100\% = 34.92\%$$

Se muestra entonces, que un voltaje de entrada de 8V, representa un 34.92% de nivel voltaje necesario para funcionamiento del circuito. También se debe mencionar que este proceso que permite determinar el porcentaje de nivel de voltaje necesario para funcionamiento de módulo de temperatura y humedad relativa, se introduce en el programa principal del sistema de procesamiento y presentación previamente definidos en la sección 3.2.2.6.

### 3.2.5. ESQUEMA TOTAL DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA

La distribución de pines del microcontrolador ATmega164P se observa en la Tabla 3.16, y el esquema total del circuito correspondiente al módulo de temperatura y humedad relativa se muestra en la Figura 3.39.

**Tabla 3.16 Asignación de Pines del Microcontrolador ATmega164P**

Número de Pin	Puerto	Descripción	Número de Pin	Puerto	Descripción
Pin 1	PB0	Sin conexión	Pin 22	PC0	Conexión al pin DB0 del LCD
Pin 2	PB1	Sin conexión	Pin 23	PC1	Conexión al pin DB1 del LCD
Pin 3	PB2	Sin conexión	Pin 24	PC2	Conexión al pin DB2 del LCD
Pin 4	PB3	Sin conexión	Pin 25	PC3	Conexión al pin DB3 del LCD
Pin 5	PB4	Sin conexión	Pin 26	PC4	Conexión al pin DB4 del LCD
Pin 6	PB5	Sin conexión	Pin 27	PC5	Conexión al pin DB5 del LCD
Pin 7	PB6	Sin conexión	Pin 28	PC6	Conexión al pin DB6 del LCD
Pin 8	PB7	Sin conexión	Pin 29	PC7	Conexión al pin DB7 del LCD
Pin 9	RESET	Conexión del pulsador	Pin 30	AVCC	Alimentación de 5V para el conversor ADC
Pin 10	VCC	Conexión de la alimentación de 5V del regulador LM7805	Pin 31	AGND	Conexión a tierra del conversor ADC
Pin 11	GND	Conexión a tierra	Pin 32	AREF	Referencia externa deshabilitada
Pin 12	XTAL2	Conexión al cristal de 4MHz	Pin 33	PA7	Conexión al pin E del LCD
Pin 13	XTAL1	Conexión al cristal de 4MHz	Pin 34	PA6	Conexión al pin RW del LCD
Pin 14	PD0	Conexión al pin Dout del XBee-Regulator	Pin 35	PA5	Conexión al pin RS del LCD
Pin 15	PD1	Conexión al pin Din del XBee-Regulator	Pin 36	PA4	Sin conexión
Pin 16	PD2	Sin conexión	Pin 37	PA3	Sin conexión
Pin 17	PD3	Conexión de la interrupción externa INT1	Pin 38	PA2	Conexión del pin BRN OUTPUT del sensor HMP45A
Pin 18	PD4	Sin conexión	Pin 39	PA1	Conexión al divisor de tensión
Pin 19	PD5	Sin conexión	Pin 40	PA0	Conexión del pin YEL OUTPUT del sensor HMP45A
Pin 20	PD6	Sin conexión			
Pin 21	PD7	Sin conexión			

*Fuente: Elaboración propia*

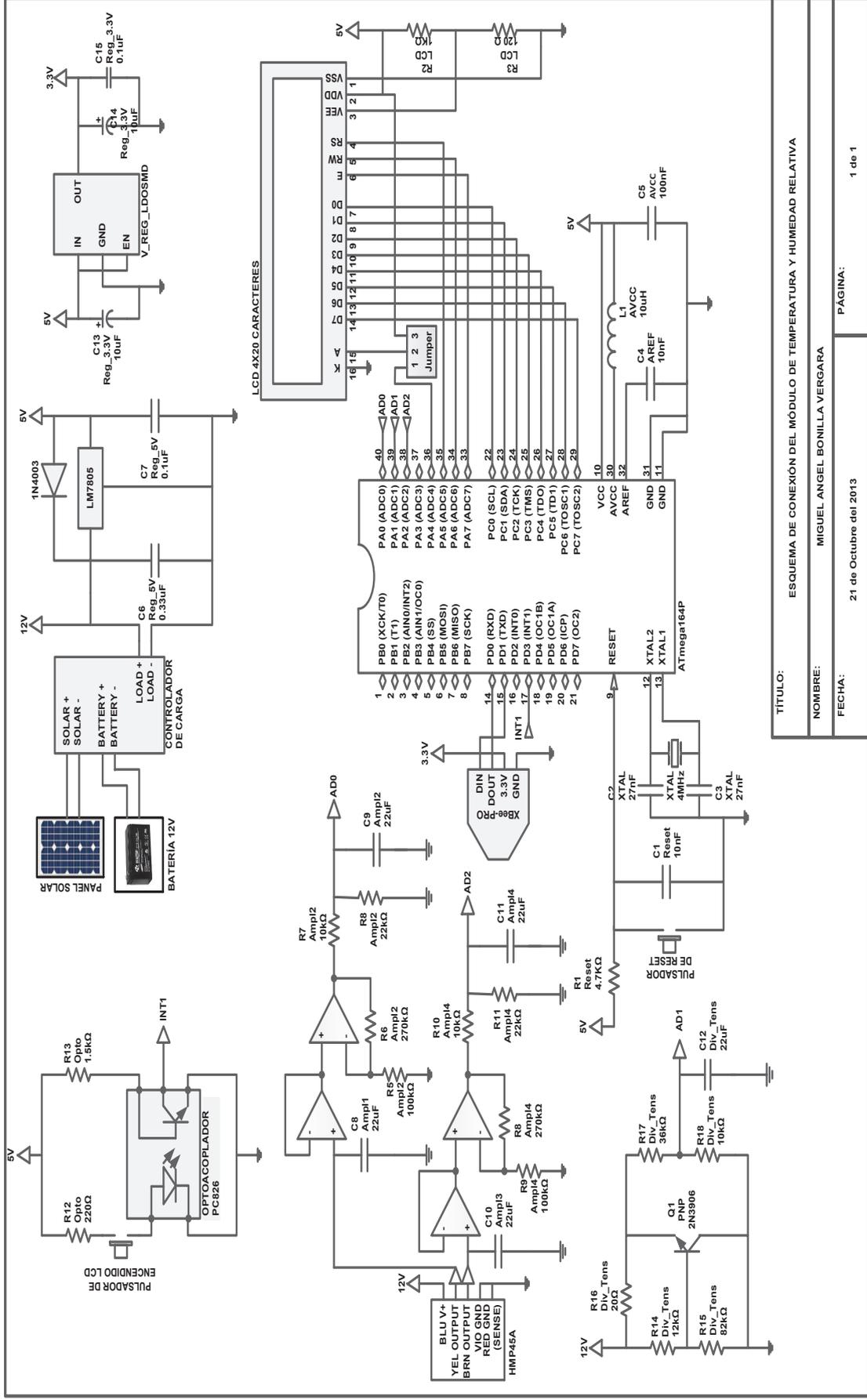


Figura 3.39 Diagrama Esquemático del Módulo de Temperatura y Humedad Relativa.  
Fuente: Elaboración propia.

TÍTULO:	ESQUEMA DE CONEXIÓN DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA
NOMBRE:	MIGUEL ANGEL BONILLA VERGARA
FECHA:	21 de Octubre del 2013
PAGINA:	1 de 1

### 3.3. DISEÑO DEL MÓDULO DE PRECIPITACIÓN

En esta sección se especifican diagramas de flujo, diagramas de bloque, estructuras de programas, circuitos y procedimientos, que permitan el diseño de cada uno de los sistemas y elementos del módulo de precipitación, tomando en cuenta que algunos de los sistemas son parecidos o iguales a los presentados y ya diseñados en el módulo de temperatura y humedad relativa.

#### 3.3.1. SISTEMAS Y ELEMENTOS DEL MÓDULO DE PRECIPITACIÓN

Tomando en cuenta el diagrama de la Figura 3.40, los sistemas del módulo de precipitación son:

- Sistema de Procesamiento y Presentación.
- Sistema de Energización.
- Sistema de Regulación y de Detección de Nivel de Voltaje.

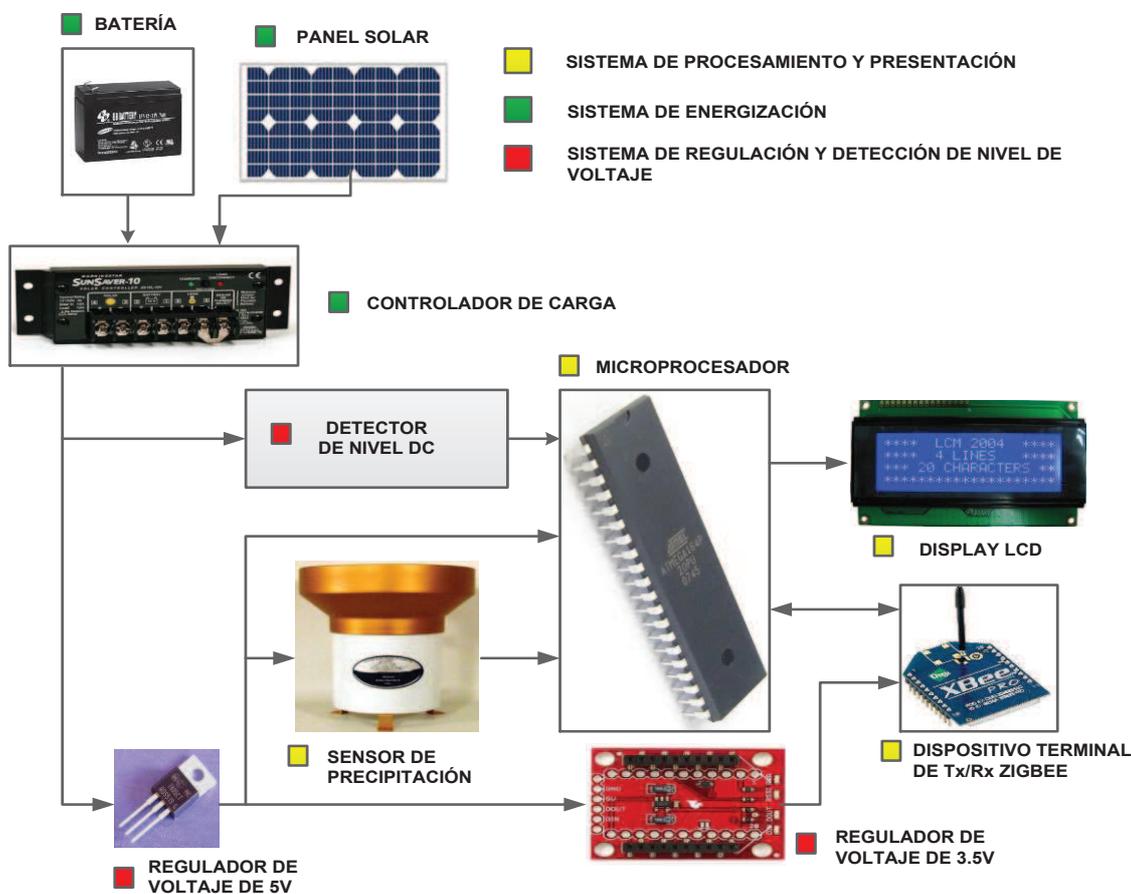


Figura 3.40 Diagrama de Bloques del Módulo de Precipitación

Fuente: Elaboración propia.

Se debe mencionar que los sistemas de energización, de regulación y detección de nivel de voltaje, son los mismos que se diseñaron para el módulo de temperatura y humedad relativa. Por ende, solo se procederá a describir el sistema de procesamiento y presentación, el cual si presenta algunos cambios con respecto al mismo sistema del módulo de temperatura y humedad relativa.

### **3.3.2. DESCRIPCIÓN Y DISEÑO DEL SISTEMA DE PROCESAMIENTO DEL MÓDULO DE PRECIPITACIÓN**

El sistema de procesamiento y presentación está conformado por el microprocesador ATmega164P, el módulo de Tx/Rx ZigBee, un display LCD de 4x20 caracteres y el sensor de precipitación TR-525M. En este sentido las funciones que debe desempeñar cada uno de los elementos de este sistema son:

#### **Microprocesador ATmega164P**

- Conversión analógica-digital de señales de voltaje del detector de nivel de voltaje, por medio del subsistema ADC que presenta el microprocesador ATmega164P.
- Muestreo de la conversión analógica-digital en un intervalo de tiempo determinado, mediante la utilización del temporizador y contador Timer/Counter1 de 16 bits del microprocesador ATmega164P.
- Detección del cambio de estado lógico que presenta el sensor de precipitación cuando hay basculaciones, por medio de la interrupción externa INT0.
- Detección de una interrupción externa INT1, que permita el encendido de la pantalla LCD
- Transmisión asincrónica de los datos digitales de precipitación al dispositivo de Tx/Rx ZigBee RFD, por medio de subsistema USART que presenta el microprocesador ATmega164P.
- Presentación de datos de nivel de voltaje de funcionamiento del módulo y datos de precipitación, mediante la utilización de las funciones de los pines de entrada/salida digitales de propósito general del ATmega164P, a los cuales se conecta un display LCD de 4x20 caracteres.

## Dispositivo de Tx/Rx ZigBee XBee-PRO

- Operación como dispositivo terminal de funciones reducidas (RFD), mediante la configuración del módulo utilizando el programa X-CTU.
- Recepción asincrónica de la información digital en caracteres de texto del microprocesador, mediante la utilización del interfaz UART.
- Transmisión inalámbrica de tramas API al dispositivo de Tx/Rx ZigBee coordinador, de la información procesada del microprocesador mediante la correspondiente configuración de dicho dispositivo.

## Display LCD

- Mostrar la información entregada por el microprocesador ATmega164P.

Se procede entonces a estructurar subprogramas que van a ser definidos como librerías de un solo programa principal, que permitan cumplir con el correspondiente procesamiento de los datos del sensor de precipitación por parte del microprocesador, tal como se muestra en la Figura 3.41(a). En el caso del dispositivo de Tx/Rx XBee-PRO, se definen parámetros de configuración, tal como se muestra en la Figura 3.41(b).

Programa de Procesamiento de Información del Módulo de Precipitación	ADC.h	Obtención de datos digitales	Procedimiento de Configuración de los Dispositivos de Tx/Rx ZigBee FFD y RFD	Configuración del interfaz serial UART
	muestreo.h	Conversión ADC en intervalos de tiempo definidos		Configuración del interfaz RF
	UART.h	Transmisión asincrónica utilizando el interfaz UART		Configuración de asociación y topología de red del dispositivo coordinador FFD
	LCD.h	Manejo de Comandos del LCD		Configuración de asociación y topología de red del dispositivo final RFD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2		Configuración del direccionamiento de origen y destino de cada dispositivo ZigBee
				Configuración del método de acceso al medio MAC

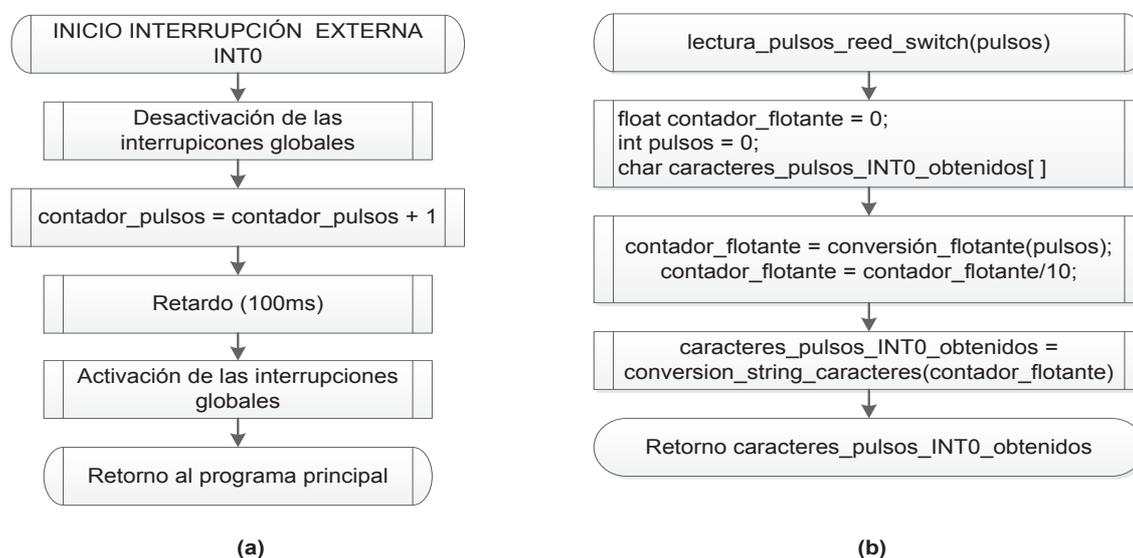
(a)

(b)

**Figura 3.41** Esquemas que muestran: (a) Conjunto de Programas del Módulo de Precipitación y (b) Conjunto de Parámetros a ser Configurados en el Dispositivo XBee-PRO RFD

Fuente: Elaboración propia

Tomando en cuenta la Figura 3.41, se debe mencionar que tanto las librerías de `muestreo.h`, `UART.h`<sup>9</sup> y `LCD.h` son las mismas que se diseñaron y describieron para el módulo de temperatura y humedad relativa. En la librería `ADC.h`<sup>10</sup> se presenta un cambio ya que solo se utiliza un solo canal para detectar señales de voltaje del detector de nivel DC, pero el proceso de configuración ADC sigue siendo el mismo. También, se introduce una nueva función de interrupción INT0 en la librería para el manejo de interrupciones externas (`interrupcion_externa.h`), que permite el conteo y lectura de pulsos de las basculaciones del sensor de precipitación y que se muestra en los diagramas de flujo de Figura 3.42.



**Figura 3.42 Diagramas de Flujo de: (a) Función de Interrupción Externa INT0 para Conteo de Basculaciones del Sensor de Precipitación y (b) Función de Lectura de Pulsos del Reed Switch del Sensor de Precipitación para Obtener Valores de Precipitación en (mm).**

*Fuente: Elaboración propia.*

### 3.3.2.1. Programa Principal de Procesamiento de Información del Módulo de Precipitación

El esquema del programa principal mostrado en la Figura 3.43<sup>11</sup>, integra las librerías de conversión ADC, muestreo, interfaz UART, manejo del LCD e

<sup>9</sup> El Anexo B.3 muestra la librería `uar.h` que se utiliza para el caso de módulo de precipitación, en el cual únicamente se hace uso de la función de transmisión.

<sup>10</sup> El Anexo B.2 contiene el desarrollo detallado de la librería `ADC.h` del módulo de precipitación, tomando en cuenta que este caso solo se procede con la lectura de un solo canal ADC.

<sup>11</sup> El programa principal del módulo de precipitación se presenta con mayor detalle en el código del Anexo B.1.

interrupciones externas, las cuales permiten realizar el procesamiento de los pulsos obtenidos del sensor de precipitación, para a continuación enviar la correspondiente información digital al dispositivo de Tx/Rx ZigBee RFD. De esta forma el esquema es el siguiente:

Programa de Procesamiento de Información del Módulo de Precipitación	ADC.h	Obtención de datos digitales
	muestreo.h	Conversión ADC en intervalos de tiempo definidos
	UART.h	Transmisión asincrónica utilizando el interfaz UART
	LCD.h	Manejo de Comandos del LCD
	Interrupcion_externa.h	Detección del cambio de transición o estado en los pines INT0, INT1 o INT2

**Figura 3.43 Programa de Procesamiento de Información dentro del Esquema del Conjunto de Programas del Módulo de Precipitación.**

*Fuente: Elaboración propia.*

1) Definición de librerías para el manejo de:

- Entradas y salidas digitales (avr/io.h).
- Subrutinas de interrupciones (avr/interrupt.h).
- Operaciones matemáticas básicas (math.h).
- Operaciones con cadenas de caracteres (string.h).
- Variables tipo enteras (stdint.h).
- Introducción de retardos de tiempo (avr/delay.h).
- Conversiones de cadenas de caracteres y valores numéricos (stdlib.h).
- Operaciones estándar de entrada y salida o presentación de datos (stdio.h).

2) Declaración de librerías previamente estructuradas y definidas como:

- Conversión ADC simple (adc.h).
- Manejo del LCD (lcd.h).
- Muestreo de la conversión ADC (muestreo.h).
- Manejo del interfaz UART (uart.h).
- Manejo de interrupciones externas (interrupción\_externa.h)

3) Declaración de variables globales, las cuales cumplen con las siguientes funciones:

- `caracteres_canal_ADC_obtenidos`, es una variable tipo `char`, que permite obtener la cadena de caracteres de los valores del sensor de precipitación.
- `contador_pulsos`, que permite contar las basculaciones del sensor de precipitación.
- `segundos_LCD`, el cual permite establecer un lapso de tiempo luego del cual la pantalla del LCD se apaga.

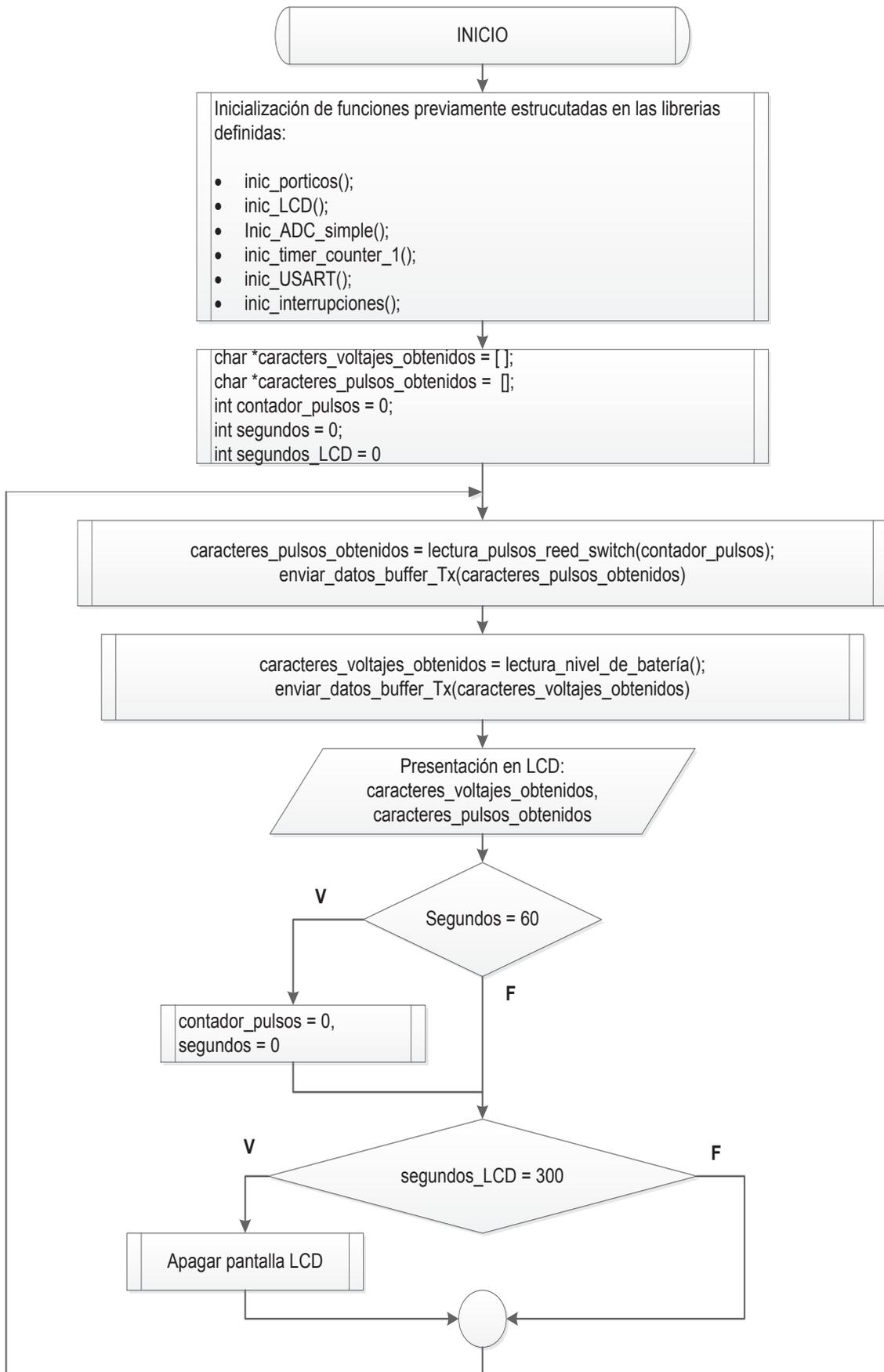
4) Definición de funciones, las cuales se enumeran como:

- `inic_porticos`, para inicialización y configuración de los pórtricos de entradas/salidas digitales de propósito general.
- `inic_interrupciones`, para inicialización de las interrupciones de los subsistemas que se requieran utilizar en el AVR.
- `lectura_nivel_bateria`, para lectura del canal AD1 en el cual se encuentra conectado el detector de nivel de voltaje necesario para funcionamiento del circuito.

5) Función principal representada mediante el diagrama de flujo de la Figura 3.44.

6) Función de inicialización de pórtricos para configuración de los diferentes pórtricos A, B, C y D de entradas/salidas digitales de propósito general, para cumplir con las siguientes tareas:

- Salidas de datos al LCD mediante la utilización de los pines PC0-PC7.
- Habilidad de los pines PA5-PA7 como salida de datos, para envío de comandos a los pines RS, RW y E respectivamente del LCD.
- Habilidad del canal de conversión ADC correspondiente al puerto AD2, mediante la utilización del pin PA2 como entrada de datos.
- Habilidad de la recepción del interfaz USART0, mediante la utilización del pin PD0 como entrada de datos.



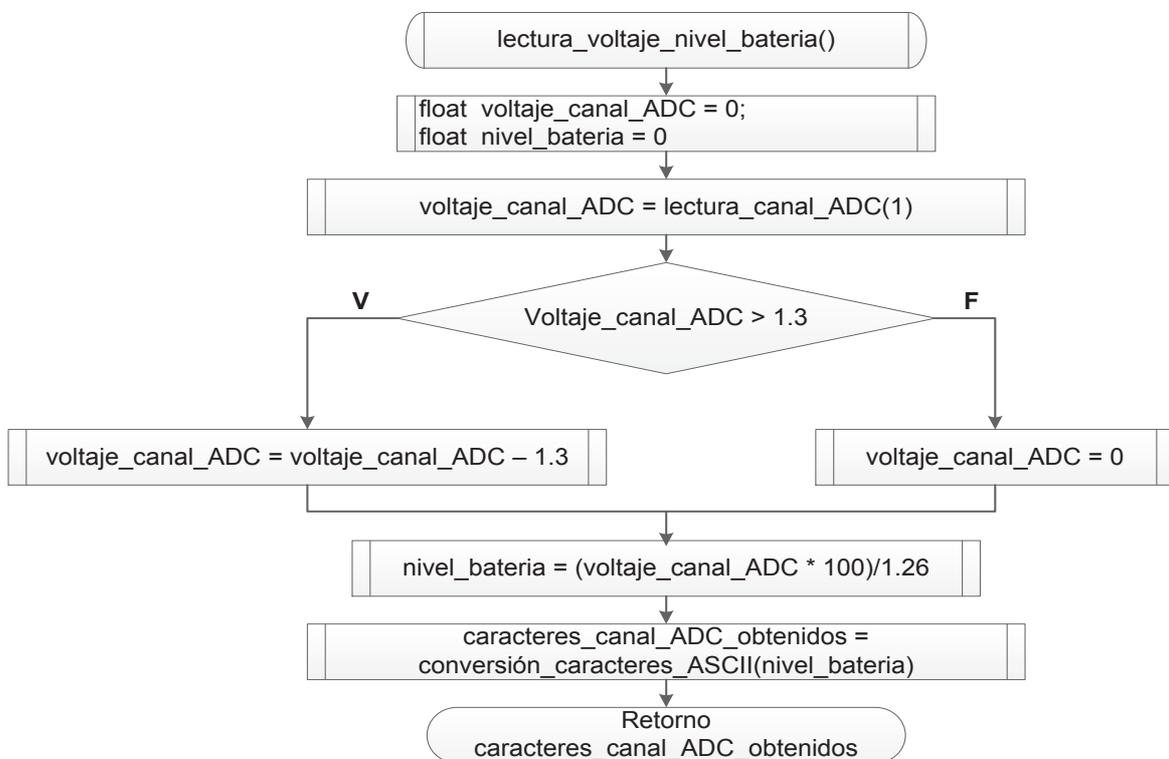
**Figura 3.44** Función Principal del Procesamiento de Señales del Sensor de Precipitación  
 Fuente: Elaboración propia

- Habilitación de la transmisión del interfaz USART0, mediante la utilización de pin PD1 como salida de datos.
- Habilitación del pin PD2 y PD3 correspondientes a las interrupciones externas INT0 e INT1 respectivamente, como entrada de datos.
- Activación de los pull-ups del resto de pines que no se utilizan.

7) Función de inicialización de las interrupciones, para habilitación de:

- Interrupción ADC.
- Interrupción de la detección de comparación de salida A y B del Timer/Counter1.
- Interrupción de recepción completa del interfaz USART0.
- Interrupción externa INT0 e INT1, cuando se detecta una transición positiva.

8) Función de lectura del nivel de voltaje necesario para funcionamiento del circuito, descrito el diagrama de flujo de la Figura 3.45.



**Figura 3.45** Función de la Lectura del Nivel de Voltaje Necesario para Funcionamiento del Circuito.

*Fuente: Elaboración propia*

9) Fin del Programa.

### 3.3.2.2. Configuración del Dispositivo de Tx/Rx ZigBee RFD del Módulo de Precipitación

Los parámetros de configuración del dispositivo terminal ZigBee RFD del módulo de precipitación se observan en la Tabla 3.17.

**Tabla 3.17 Parámetros y Valores de Configuración del Dispositivo RFD ZigBee del Módulo de Precipitación**

Configuración	Parámetros	Comando	Valor del Parámetro	Descripción
Interfaz Serial UART	Velocidad de transmisión	ATBD	3	9600bps
	Paridad	ATNB	0	No-paridad
	Tiempo de retardo	ATRO	0x0F	Número de caracteres almacenados en el buffer antes de ser enviados
	Operación API	ATAP	0	Des-habilitación de la recepción de tramas API del módulo coordinador
Interfaz RF	Nivel de Potencia	ATPL	4	Potencia de transmisión de 18dBm
	Umbral CCA	ATCA	0x2C	Umbral de -44dBm sobre el cual se descartan paquetes de recepción
Topología y Asociación	Tipo de Dispositivo	ATCE	0	Dispositivo final
	Tipo de asociación con el módulo coordinador	ATA1	Bit-0 = 0	Asociación con un dispositivo coordinador cuyo PAN ID es determinado por el identificador del nodo
	Canal de operación		Bit-1 = 1	Asociación con un dispositivo coordinador que opera en cualquier canal disponible
	Auto-asociación		Bit-2 = 1	El dispositivo intentará asociarse hasta tener éxito
	Operación de poleo		Bit-3 = 0	No se realizará operación de poleo luego de la activación del pin
Dirección de origen	ATMY		0xFFFF	Utilización del serial del módulo como dirección de origen: SL = 0x406CCCBE, SH = 0x13A200
Direccionamiento	Dirección de destino (32 bits más significativos)	ATDH	0x13A200	El módulo coordinador solo utilizará 32 bits menos significativos de direccionamiento de destino
	Dirección de destino (32 bits menos significativos)	ATDL	0x40710AB9	Dirección de destino del módulo coordinador
	Dirección ID de la red PAN	ATID	0x3332	Igual dirección que el módulo coordinador
Método de Acceso al Medio CSMA-CA	Valor del exponente de Back-off	ATRN	1	Valor mínimo del exponente de Back-off = (valor entero entre 0 y $2^{BE} - 1$ )x (aUnitBackoffPeriod)
	Número máximo de reintentos	ATRR	0	Se ejecutará solo 3 reintentos especificados por el estándar IEEE 802.15.4

Fuente: Elaboración propia basada en la sección 3.2.2.7.

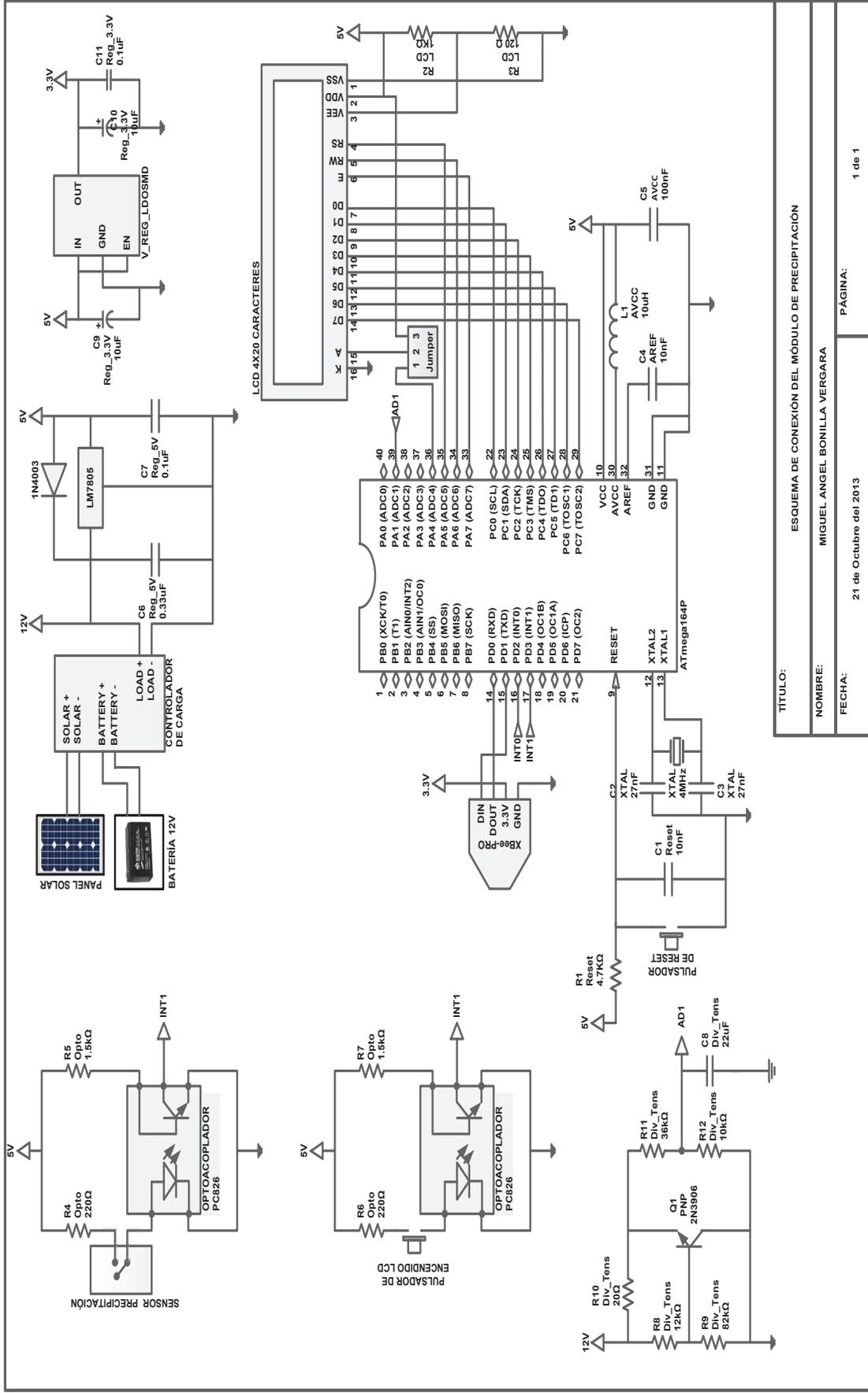
### 3.3.3. ESQUEMA TOTAL DEL MÓDULO DE PRECIPITACIÓN

La distribución de pines del microcontrolador ATmega164P para este módulo, se observa en la Tabla 3.18, y el esquema total del circuito correspondiente al módulo de precipitación se muestra en la Figura 3.46.

**Tabla 3.18 Asignación de Pines del Microcontrolador ATmega164P del Módulo de Precipitación**

Número de Pin	Puerto	Descripción	Número de Pin	Puerto	Descripción
Pin 1	PB0	Sin conexión	Pin 22	PC0	Conexión al pin DB0 del LCD
Pin 2	PB1	Sin conexión	Pin 23	PC1	Conexión al pin DB1 del LCD
Pin 3	PB2	Sin conexión	Pin 24	PC2	Conexión al pin DB2 del LCD
Pin 4	PB3	Sin conexión	Pin 25	PC3	Conexión al pin DB3 del LCD
Pin 5	PB4	Sin conexión	Pin 26	PC4	Conexión al pin DB4 del LCD
Pin 6	PB5	Sin conexión	Pin 27	PC5	Conexión al pin DB5 del LCD
Pin 7	PB6	Sin conexión	Pin 28	PC6	Conexión al pin DB6 del LCD
Pin 8	PB7	Sin conexión	Pin 29	PC7	Conexión al pin DB7 del LCD
Pin 9	RESET	Conexión del pulsador	Pin 30	AVCC	Conexión a la alimentación de 5V para el conversor ADC
Pin 10	VCC	Conexión de la alimentación de 5V del regulador LM7805	Pin 31	AGND	Conexión a tierra del conversor ADC
Pin 11	GND	Conexión a tierra	Pin 32	AREF	Referencia externa deshabilitada
Pin 12	XTAL2	Conexión al cristal de 4MHz	Pin 33	PA7	Conexión al pin E del LCD
Pin 13	XTAL1	Conexión al cristal de 4MHz	Pin 34	PA6	Conexión al pin RW del LCD
Pin 14	PD0	Conexión al pin Dout del XBee-Regulator	Pin 35	PA5	Conexión al pin RS del LCD
Pin 15	PD1	Conexión al pin Din del XBee-Regulator	Pin 36	PA4	Sin conexión
Pin 16	PD2	Conexión de la interrupción externa INT0	Pin 37	PA3	Sin conexión
Pin 17	PD3	Conexión de la interrupción externa INT1	Pin 38	PA2	Sin conexión
Pin 18	PD4	Sin conexión	Pin 39	PA1	Conexión al divisor de tensión
Pin 19	PD5	Sin conexión	Pin 40	PA0	Sin conexión
Pin 20	PD6	Sin conexión			
Pin 21	PD7	Sin conexión			

*Fuente: Elaboración propia*



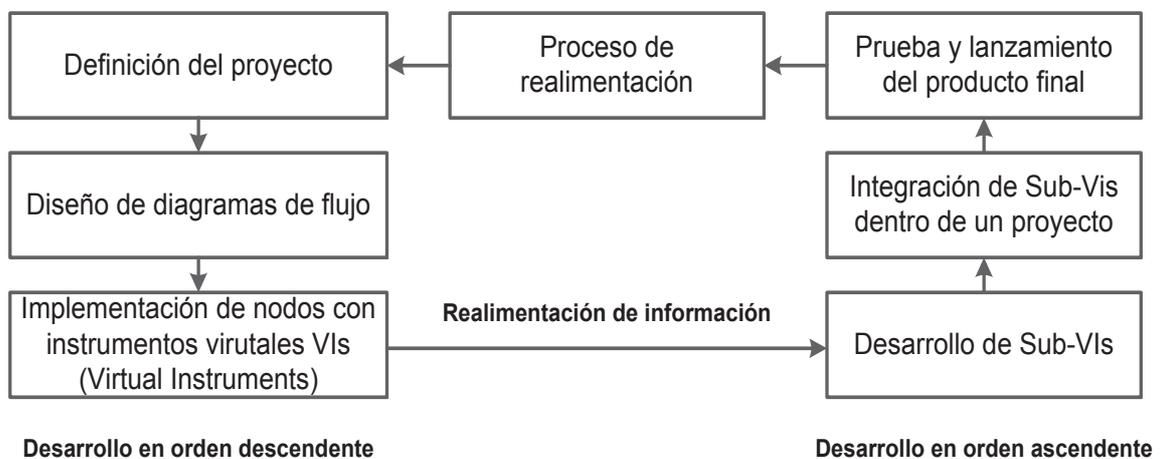
TÍTULO:	ESQUEMA DE CONEXIÓN DEL MÓDULO DE PRECIPITACIÓN
NOMBRE:	MIGUEL ANGEL BONILLA VERGARA
FECHA:	21 de Octubre del 2013
PÁGINA:	1 de 1

Figura 3.46 Diagrama Esquemático del Módulo de Precipitación.  
Fuente: Elaboración propia.

### 3.4. DISEÑO DEL MÓDULO DE ADQUISICIÓN DE DATOS [36]

La adquisición de datos se realiza por medio de un conjunto de programas desarrollados en LabView 2010. Se debe entender que cada sub-programa se ha desarrollado de forma individual para luego ser integrados en uno solo programa, que facilite al usuario el control y gestión de los datos meteorológicos obtenidos tanto del módulo de temperatura y humedad relativa, como del módulo de precipitación previamente diseñados.

Inicialmente se debe definir características generales y especificaciones del proyecto. A continuación, se define tareas específicas para cumplir con las especificaciones generales. Luego de haber definido las tareas que las aplicaciones deben realizar, se procede a desarrollar subVIs<sup>12</sup> que represente el periodo de desarrollo ascendente de la Figura 3.47. El proceso de realimentación una vez que se haya concretado un producto, es importante para mejorar y establecer nuevas versiones del proyecto final.



**Figura 3.47 Diagrama de Bloques del Proceso de Diseño del Módulo de Adquisición de Datos**

*Fuente: Elaboración propia basado en [36]*

En esta sección se observarán diagramas de flujo y de bloques que describen el funcionamiento de cada subVI, los cuales especifican la implementación y diseño

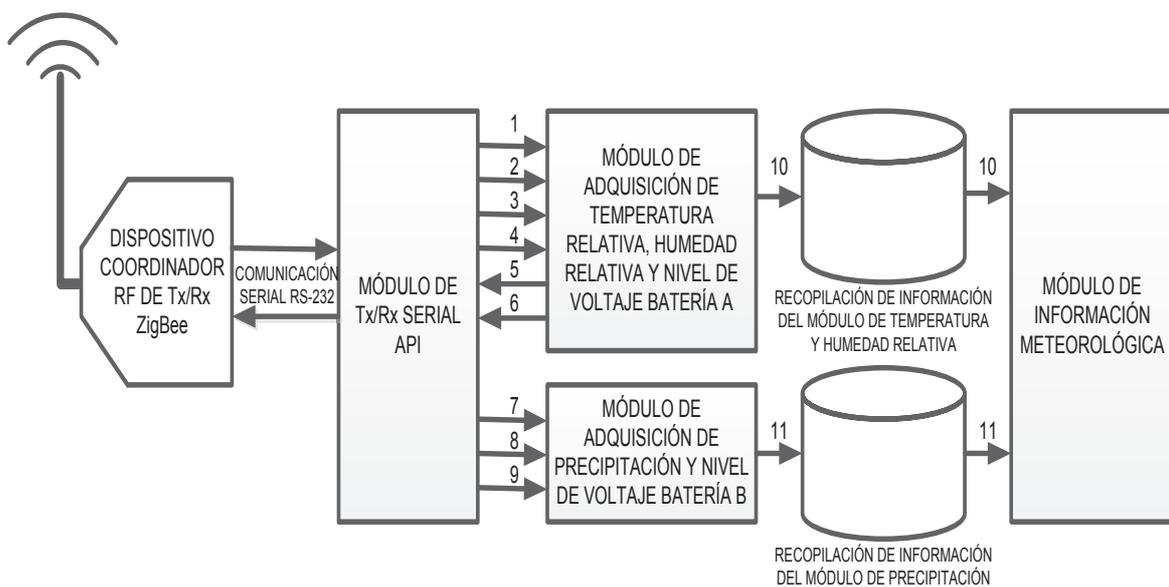
<sup>12</sup> Se entiende por Vi al instrumento virtual que permite realizar el manejo y observación por parte del usuario, de la ejecución de un determinado programa en LabView. Por otro lado, un SubVi es una subrutina de que se ejecuta dentro de un Vi.

de cada uno de los sistemas y elementos en el que el módulo de adquisición de datos ha sido dividido. Se debe mencionar que el diagrama de bloques total de cada uno de los sistemas se muestra en el Anexo C.

### 3.4.1. SISTEMAS Y ELEMENTOS DEL MÓDULO DE ADQUISICIÓN DE DATOS

La Figura 3.48 muestra los sistemas y elementos que integran a este módulo, los cuales son:

- a) Módulo Coordinador RF de Tx/Rx ZigBee FFD.
- b) Módulo de Tx/Rx Serial API.
- c) Módulo de Adquisición de Temperatura Relativa, Humedad Relativa y Nivel de Voltaje de Batería A.
- d) Módulo de Adquisición de Precipitación y Nivel de Voltaje de Batería B.
- e) Módulo de Presentación de Datos.



**Figura 3.48** Sistemas y Elementos del Módulo de Adquisición de Datos

*Fuente: Elaboración propia*

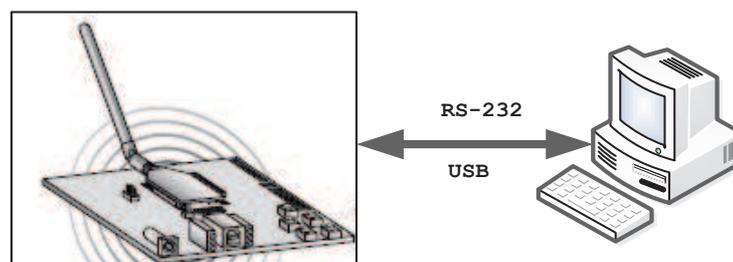
Se observa que en la entrada y salida de cada uno de los módulos se presentan una serie de señales que se describen a continuación en forma secuencial:

- 1) Cadena de caracteres del voltaje del sensor de temperatura relativa.
- 2) Cadena de caracteres del voltaje del sensor de humedad relativa.

- 3) Cadena de caracteres del nivel de voltaje de la batería A en porcentaje.
- 4) Cadena de caracteres del nivel de señal RSSI del módulo de Tx/Rx ZigBee del módulo de temperatura y humedad relativa.
- 5) Valor de temperatura relativa procesada para ser enviada al módulo coordinador FFD.
- 6) Valor de humedad relativa procesada para ser enviada al módulo coordinador FFD.
- 7) Cadena de caracteres del valor de basculaciones del sensor de precipitación.
- 8) Cadena de caracteres del nivel de voltaje de la batería B en porcentaje.
- 9) Cadena de caracteres del nivel de señal RSSI del módulo de Tx/Rx ZigBee del módulo de precipitación.
- 10) Información procesada del módulo de temperatura y humedad relativa: fecha, hora, valor de temperatura relativa, valor de temperatura promedio, valor de humedad relativa, valor de humedad promedio, nivel de voltaje de batería A, estado sensor temperatura, estado sensor humedad y estado de nivel de voltaje de batería A.
- 11) Información procesada del módulo de precipitación: fecha, hora, valor de precipitación, nivel de voltaje de batería B, estado sensor precipitación, y estado de nivel de voltaje de batería B.

### 3.4.2. DESCRIPCIÓN DEL DISPOSITIVO DE TX/RX ZIGBEE FFD

El módulo coordinador ZigBee FFD, tiene como propósito recibir la información de del módulo de precipitación y del módulo de temperatura y humedad relativa, para reenviar dicha información por medio de un interfaz RS-232 o USB al adquisidor de datos en formatos de tramas API, tal como se observa en la Figura 3.49.



**Figura 3.49** Comunicación entre el Adquisidor de Datos y el Módulo de Tx/Rx ZigBee FFD

*Fuente: Elaboración propia*

De igual forma, el dispositivo coordinador recibe información procesada de los valores de temperatura y humedad relativa en formatos de tramas API, para transmitir al módulo de temperatura y humedad relativa, el cual se encargará de presentar dichos valores en el LCD.

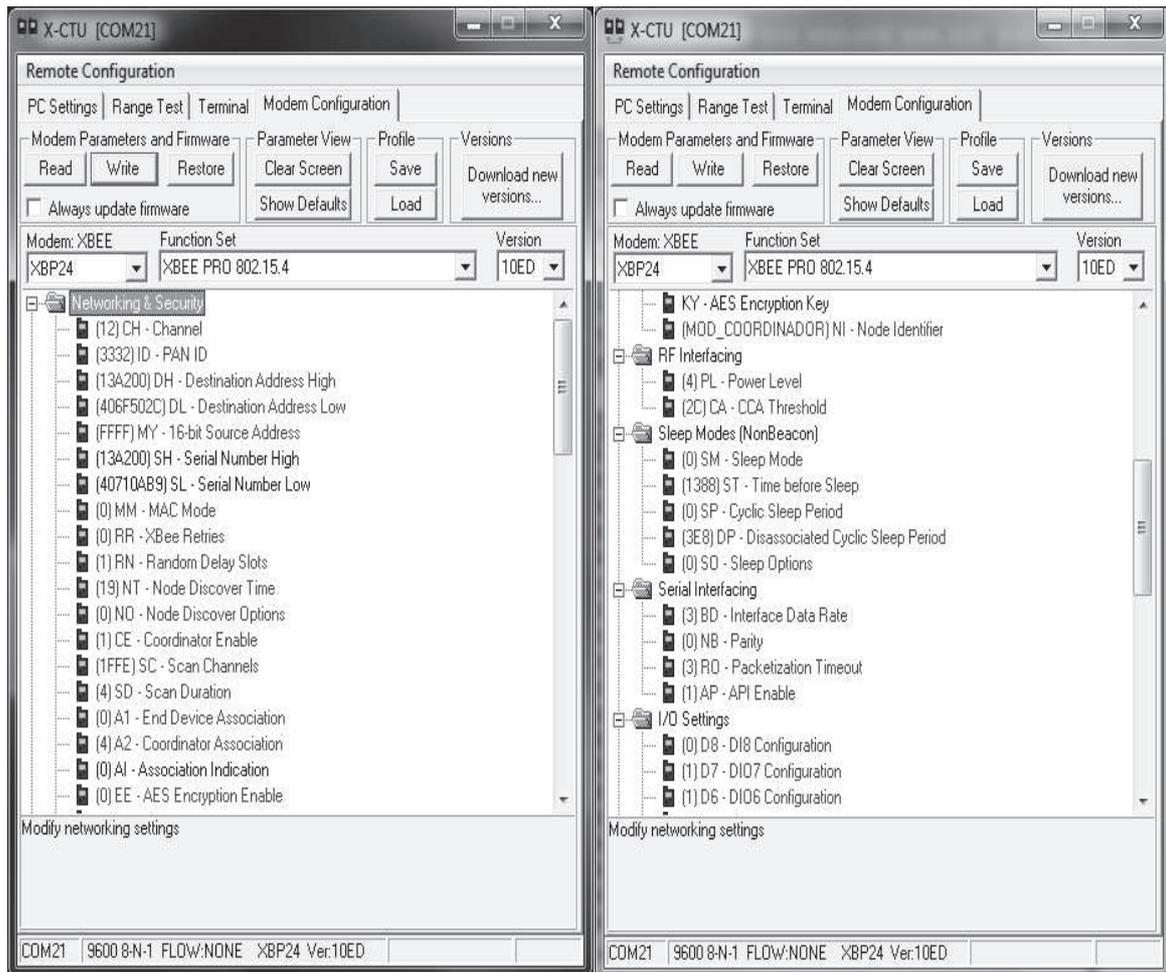
### 3.4.2.1. Parámetros y Valores de Configuración del Módulo de Tx/Rx ZigBee FFD del Módulo de Adquisición de Datos

**Tabla 3.19 Parámetros y Valores de Configuración del Módulo FFD ZigBee Coordinador del Módulo de Adquisición de Datos**

Configuración	Parámetros	Comando	Valor del Parámetro	Descripción
Interfaz Serial UART	Velocidad de transmisión	ATBD	3	9600bps
	Paridad	ATNB	0	No-paridad
	Tiempo de retardo	ATRO	0x03	Número de caracteres almacenados en el buffer antes de ser enviados
	Operación API	ATAP	1	Habilitación de la recepción y transmisión de tramas API del módulo coordinador
Interfaz RF	Nivel de Potencia	ATPL	4	Potencia de transmisión de 18dBm
	Umbral CCA	ATCA	0x2C	Umbral de -44dBm sobre el cual se descartan paquetes de recepción
Topología y Asociación	Tipo de Dispositivo	ATCE	1	Dispositivo coordinador
	Modo de exploración activa del módulo coordinador	ATA2	Bit-0 = 0	Define si el módulo coordinador realiza censado del canal de comunicaciones en busca de una dirección PAN ID mediante la activación de este bit en 1L, caso contrario, se debe establecer la dirección ID de la red PAN para comunicación con otros módulos RF que presenten la misma dirección
	Detección de nivel de energía		Bit-1 = 0	Activación de censado del canal de comunicaciones para detección de cualquier canal libre mediante la activación de este bit en 1L, caso contrario, se debe especificar el canal RF de operación
	Permisos de asociación		Bit-2 = 1	Permite la asociación de un dispositivo final RFD al módulo coordinador mediante la activación de este bit 1L
Direccionamiento	Dirección de origen	ATMY	0xFFFE	Utilización del serial del módulo como dirección de origen: SL = 0x40710AB9, SH = 0x13A200
	Dirección de destino (32 bits más significativos)	ATDH	0x13A200	32 bits más significativos de la dirección de destino del módulo de temperatura y humedad relativa
	Dirección de destino (32 bits menos significativos)	ATDL	0x406F502C	32 bits menos significativos de la dirección de destino del módulo de temperatura y humedad relativa
	Dirección ID de la red PAN	ATID	0x3332	Igual dirección que el módulo coordinador
Método de Acceso al Medio CSMA-CA	Valor del exponente de Back-off	ATRN	1	Valor mínimo del exponente de Back-off = (valor entero entre 0 y $2^{BE} - 1$ )x (aUnitBackoffPeriod)
	Número máximo de reintentos	ATRR	0	Se ejecutará solo 3 reintentos especificados por el estándar IEEE 802.15.4

Fuente: Elaboración propia basado en la sección 3.2.2.7.

Tomando en cuenta el procedimiento de configuración de un dispositivo ZigBee de la sección 3.2.2.7, se presenta en la Tabla 3.19 los valores y parámetros que se configurarán para el caso del dispositivo coordinador XBee-PRO FFD del módulo de adquisición de datos. De igual forma, se presenta en la Figura 3.50 los diferentes parámetros de configuración del presente módulo utilizando el programa XCTU.



**Figura 3.50** Ventanas del Software X-CTU que Muestran los Parámetros Configurados del Dispositivo Coordinador FFD ZigBee del Módulo de Adquisición de Datos

Fuente: X-CTU.

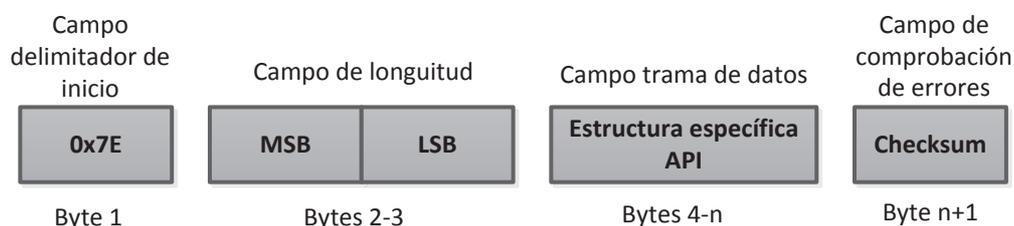
### 3.4.3. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE RECEPCIÓN Y TRANSMISIÓN SERIAL API

Este sistema tiene como propósito recibir información en formatos de tramas API del módulo coordinador ZigBee FFD, para procesar y extraer del campo de datos RF de cada una de las tramas, la información de los sensores de temperatura, humedad, precipitación, y además nivel de señal RSSI y nivel de voltaje de

batería de cada uno de los módulos. De igual manera, este sistema envía los valores de temperatura y humedad muestreados y procesados al dispositivo coordinador ZigBee FFD.

Como una alternativa a la operación transparente, la cual actúa como una línea de transmisión serial, aparece la operación de interface de programación de aplicaciones (*Application Programming Interface API*), que requiere que la comunicación con un determinado módulo sea a través de un interfaz, en el cual los datos son enviados en tramas en un orden definido. El API especifica como los comandos, comandos de respuestas y mensajes de estado del módulo son enviados y recibidos desde el dispositivo ZigBee, usando una trama de datos UART [28].

Una vez que se habilita la recepción y transmisión de tramas API sin caracteres de espacio en un determinado dispositivo ZigBee, la trama de datos UART se define tal como se muestra en la Figura 3.51.



**Figura 3.51 Formato de la Trama API sin Caracteres de Espacio**

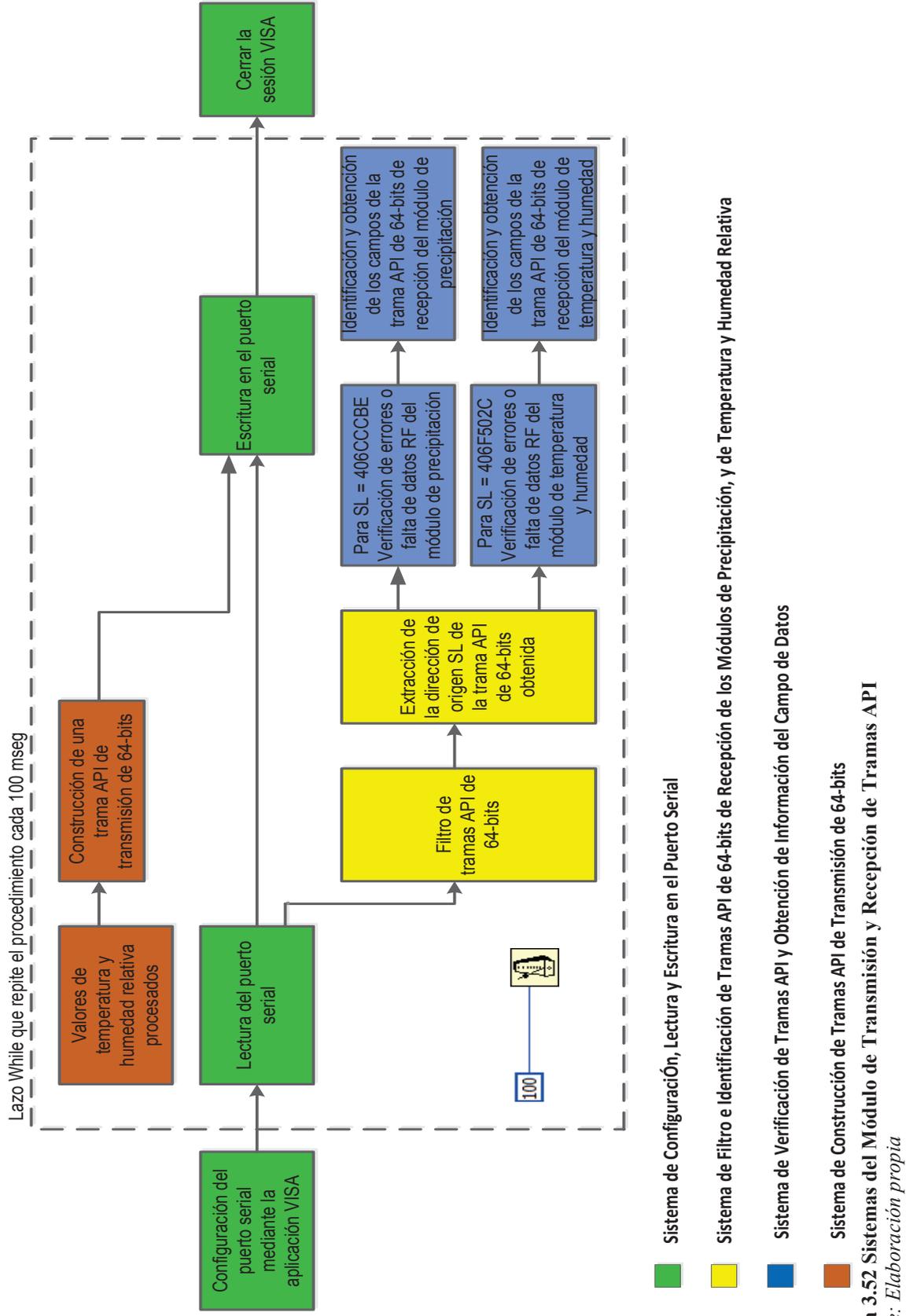
*Fuente: Elaboración propia basado en [28]*

La Figura 3.52<sup>13</sup> muestra como este módulo es dividido en cuatro sistemas, los cuales van a ser descritos en las secciones subsiguientes.

#### **3.4.3.1. Sistema de Configuración, Lectura y Escritura en el Puerto Serial**

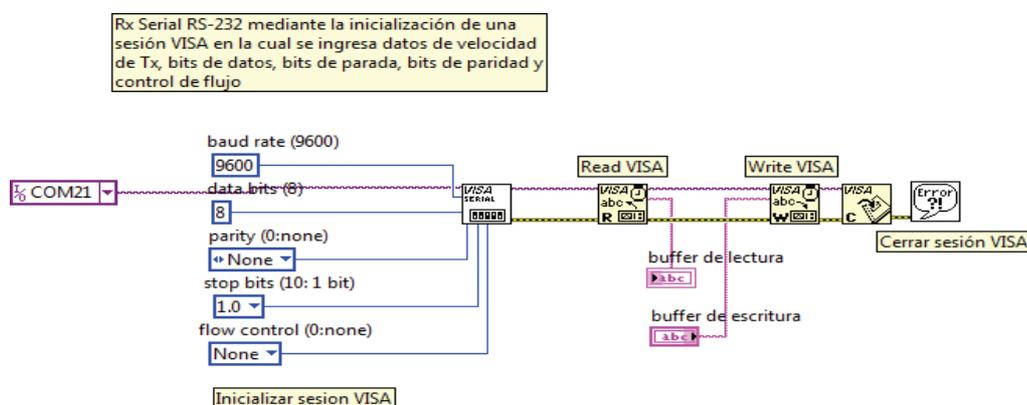
La arquitectura de software de un instrumento virtual (*Virtual Instrument Software Architecture VISA*), es un estándar para configuración, programación y resolución

<sup>13</sup> El Anexo C1 presenta el diagrama de bloques detallado del desarrollo de este módulo de transmisión y recepción serial API.



de problemas de instrumentación que comprende los interfaces GPIB, VXI, PXI, Serial, Ethernet y USB. VISA provee el interfaz de comunicación serial para programación entre el hardware y la aplicación desarrollada en LabView. Para utilizar la aplicación VISA en LabView, es necesario que el correspondiente driver sea instalado en el computador [37].

La utilización del estándar VISA en LabView mostrado en la Figura 3.53, requiere de los siguientes pasos:



**Figura 3.53 Diagrama de Bloques del Interfaz VISA de LabView**

Fuente: Elaboración propia utilizando el software LabView 2010

- 1) Apertura de una sesión VISA en la cual se configura los parámetros mostrados en la Tabla 3.20.

**Tabla 3.20 Configuración del Interfaz VISA para Comunicación Serial con el Módulo ZigBee Coordinador**

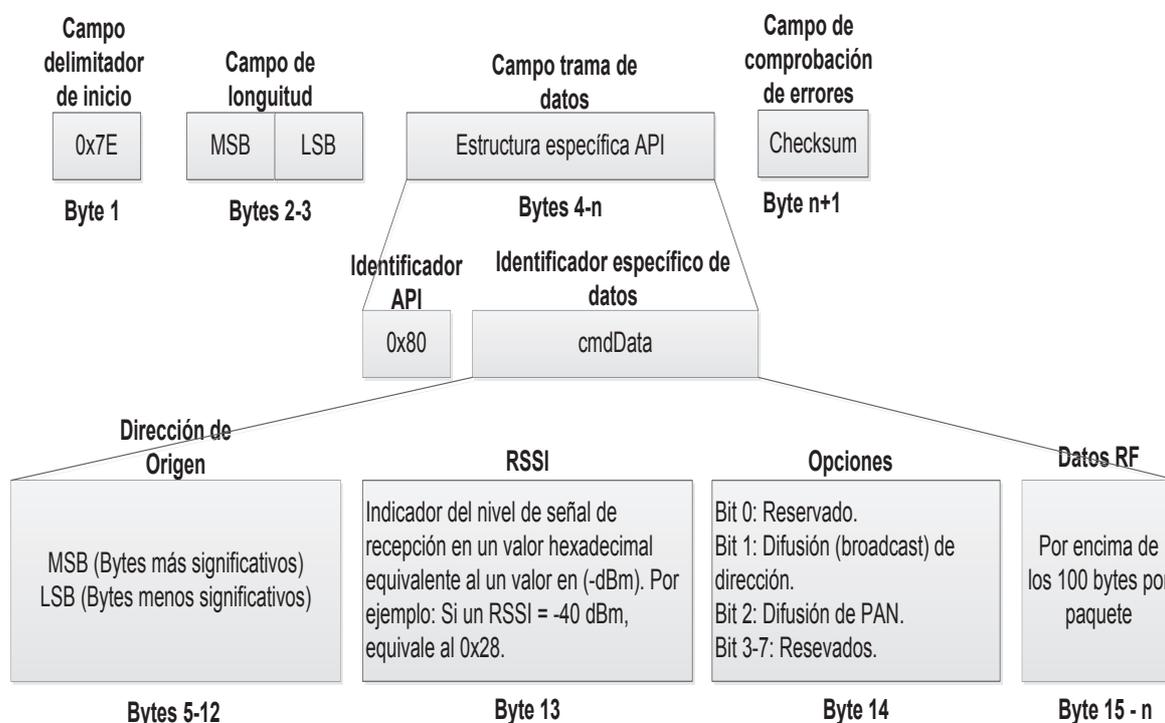
Parámetro	Descripción
Velocidad de transmisión	9600bps
Puerto serial de entrada/salida	COMx, donde x depende del puerto del computador
Bits de datos	8
Paridad	Ninguna
Bits de parada	1
Control de flujo	Ninguno

Fuente: Elaboración propia

- 2) Lectura del puerto serial mediante el bloque VISA read.
- 3) Escritura en el puerto serial mediante el bloque VISA write.
- 4) Cierre de la sesión VISA.

### 3.4.3.2. Sistema de Filtro e Identificación de Campos de una Trama API de Recepción de 64-bits

Este sistema tiene como propósito filtrar información para obtener tramas API de recepción de 64-bits del buffer de entrada del interfaz VISA<sup>14</sup>, que presenten un delimitador de inicio 0x7E con una longitud de bytes mayor a la longitud de una trama mínima API de 14-bytes y que a su vez concuerde con el campo de longitud, para a continuación extraer los campos que conforman esta trama<sup>15</sup>, tomando en cuenta que la trama API mencionada presenta un valor de identificación API de 0x80 y cuya estructura se muestra en la Figura 3.54.



**Figura 3.54 Trama de un Paquete de Recepción de 64-bits de Direccionamiento.**

*Fuente: Elaboración propia basado en [28]*

Bytes trama API mínima = 1 byte del delimitador de inicio +  
 2bytes del campo de longitud + 11bytes del campo de datos + (Ecuación 3.26)  
 1byte de checksum

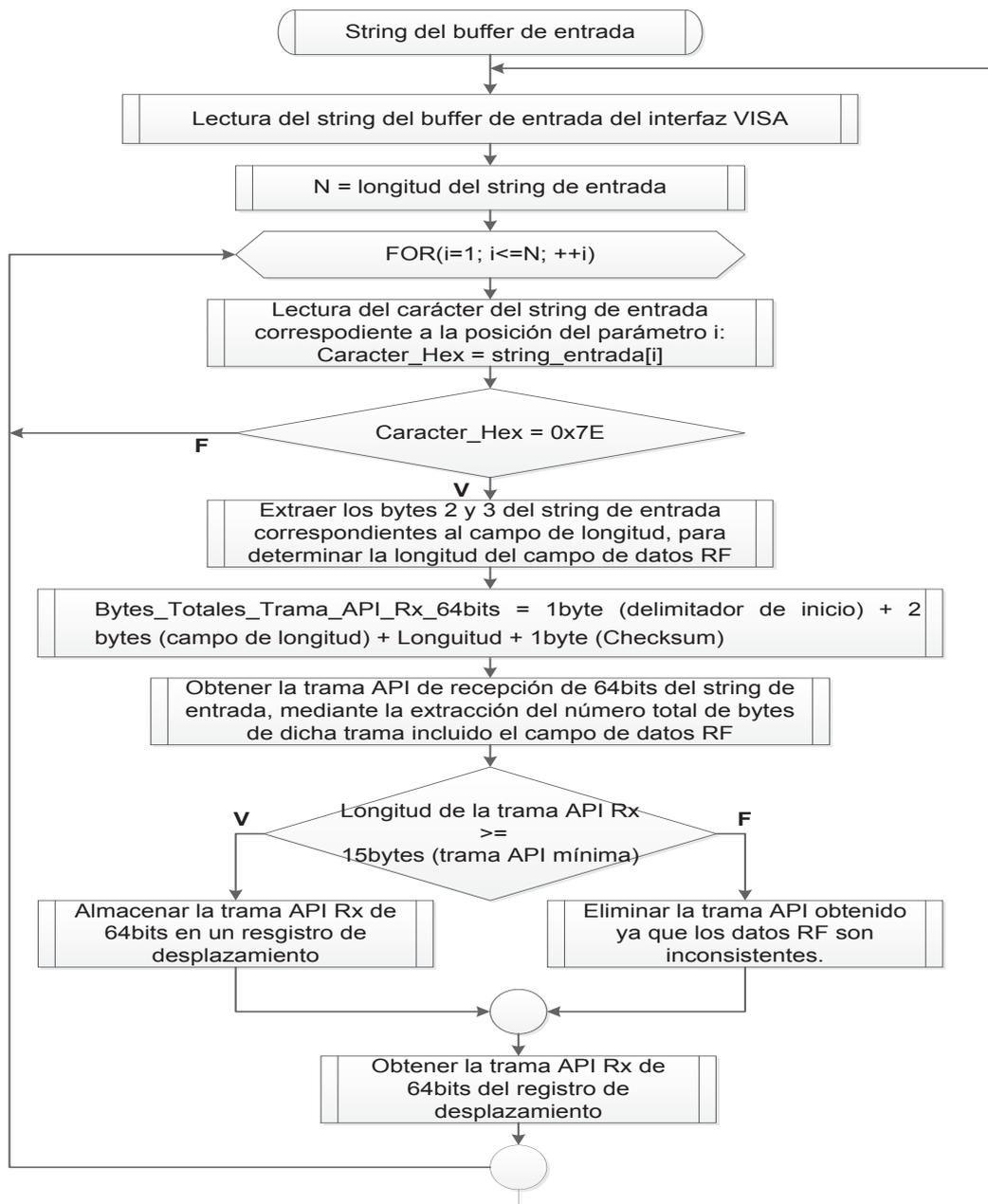
<sup>14</sup> El Anexo C1.1, presenta con detalle el diagrama de bloques del proceso de obtención de la trama API en el buffer de recepción.

<sup>15</sup> El Anexo C1.2, muestra el diagrama de bloques correspondiente a la extracción de los diferentes campos que conforman una trama API de recepción de 64-bits.

**Donde:**

11bytes del campo de datos = 1byte del identificador API +  
 8bytes de la dirección de origen + 1byte del RSSI + 1byte de (Ecuación 3.27)  
 opciones + 0bytes de datos RF

La función de este sistema se pueden observar en el diagrama de flujo de la Figura 3.55.

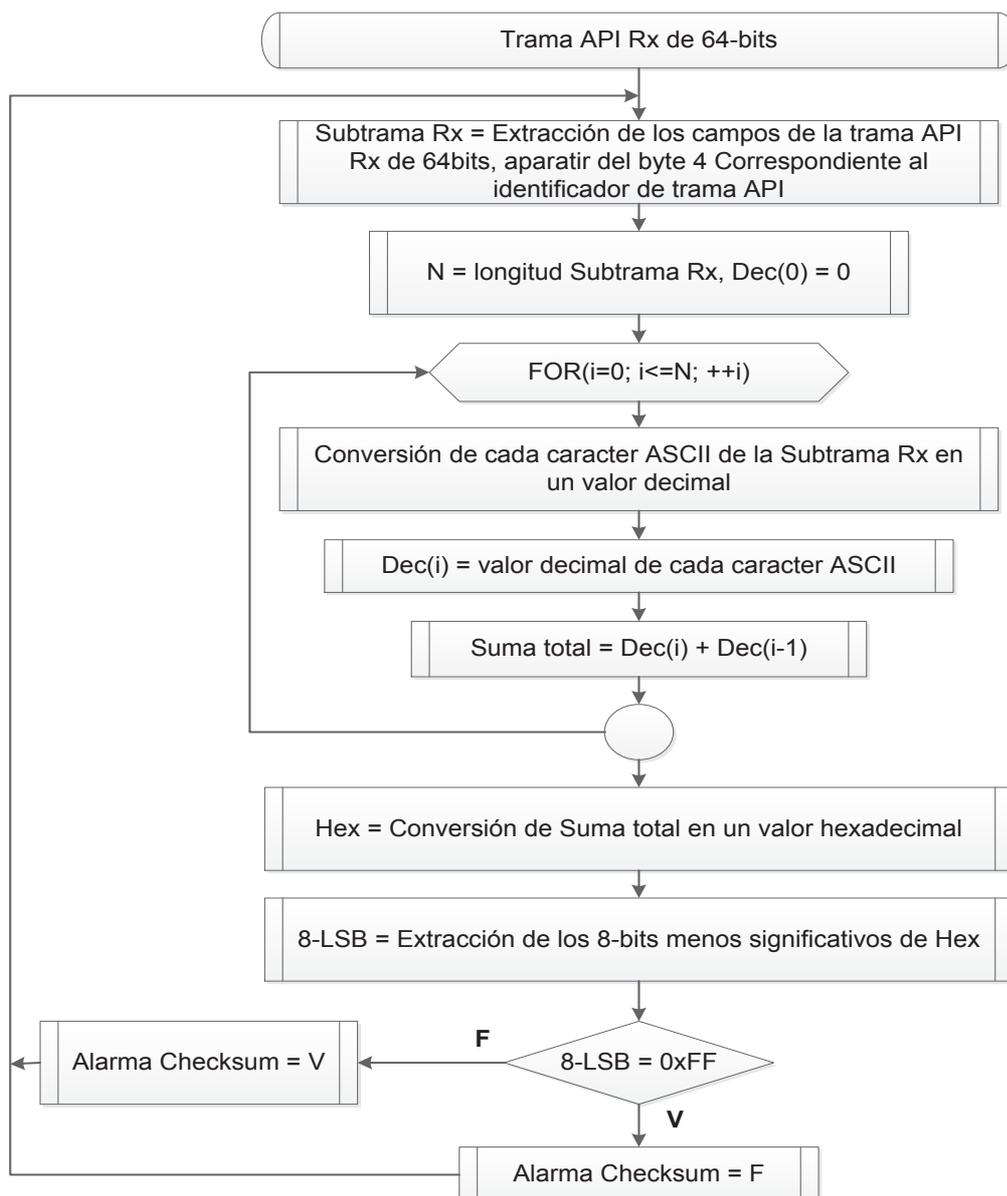


**Figura 3.55 Diagrama de Flujo de la Función de Filtrado de Tramas API Rx de 64-bits de Direccionamiento**

*Fuente: Elaboración propia*

### 3.4.3.3. Sistema de Verificación de Tramas API y Obtención de Información del Campo de Datos

Este sistema tiene como función verificar que las tramas API de recepción de 64-bits<sup>16</sup> obtenidas por el anterior sistema, no presenten errores o falta de datos RF tal como se muestra en la Figura 3.56.

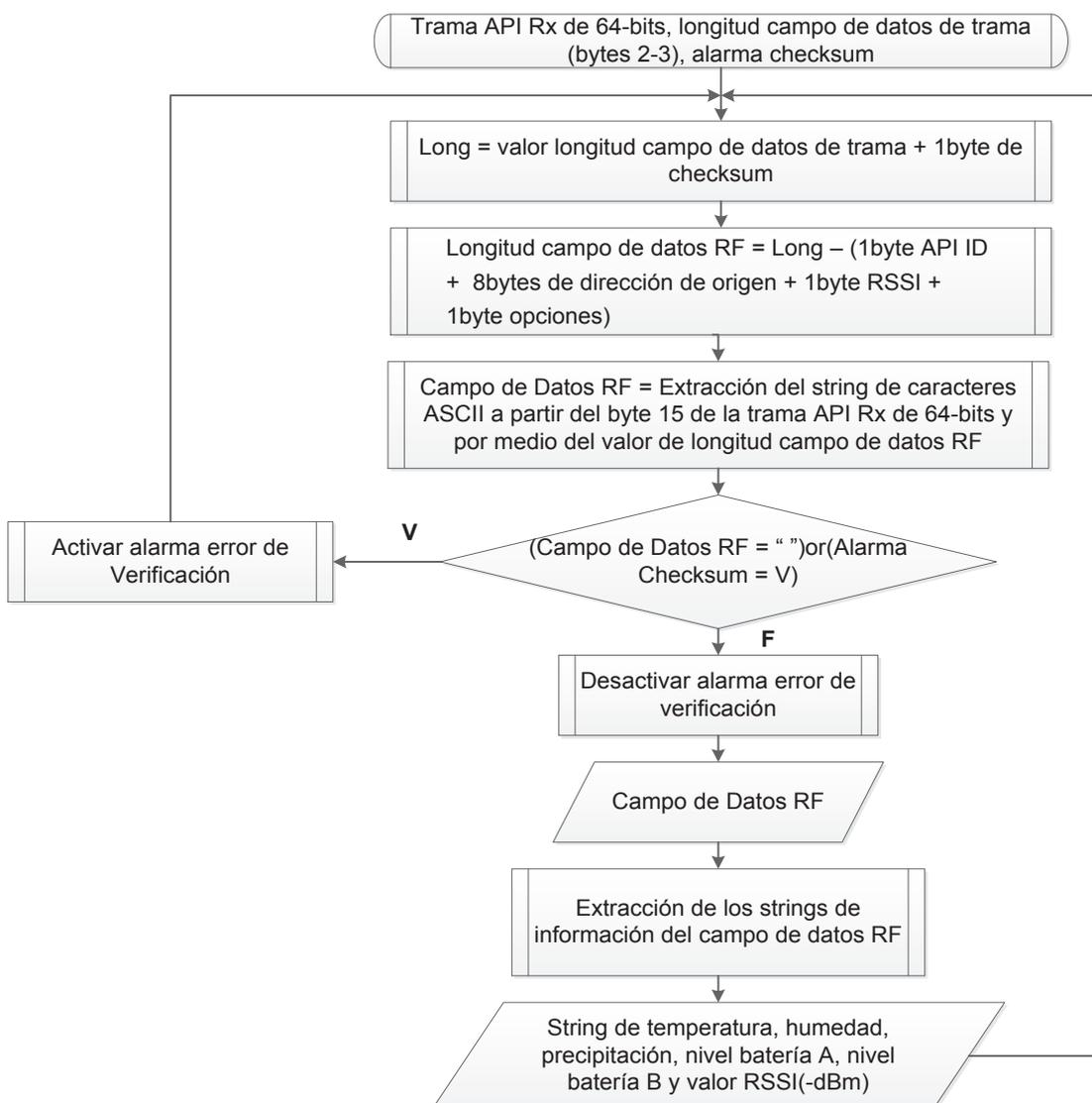


**Figura 3.56** Diagrama de Flujo de la Función de Chequeo del Campo de Checksum de la Trama API Rx de 64-bits

Fuente: Elaboración propia

<sup>16</sup> El Anexo C1.3 presenta el diagrama de bloques detallado de la verificación de una trama API de recepción.

De igual manera, una vez que se haya verificado la validez de las tramas API recibidas, se procede a extraer finalmente la cadena de datos<sup>17</sup> de temperatura, humedad, precipitación, nivel de batería A, nivel de batería B y niveles de señal RSSI; que serán almacenados en variables globales para su correspondiente procesamiento por los módulos subsiguientes, tal como se muestra en el diagrama de flujo de la Figura 3.57.



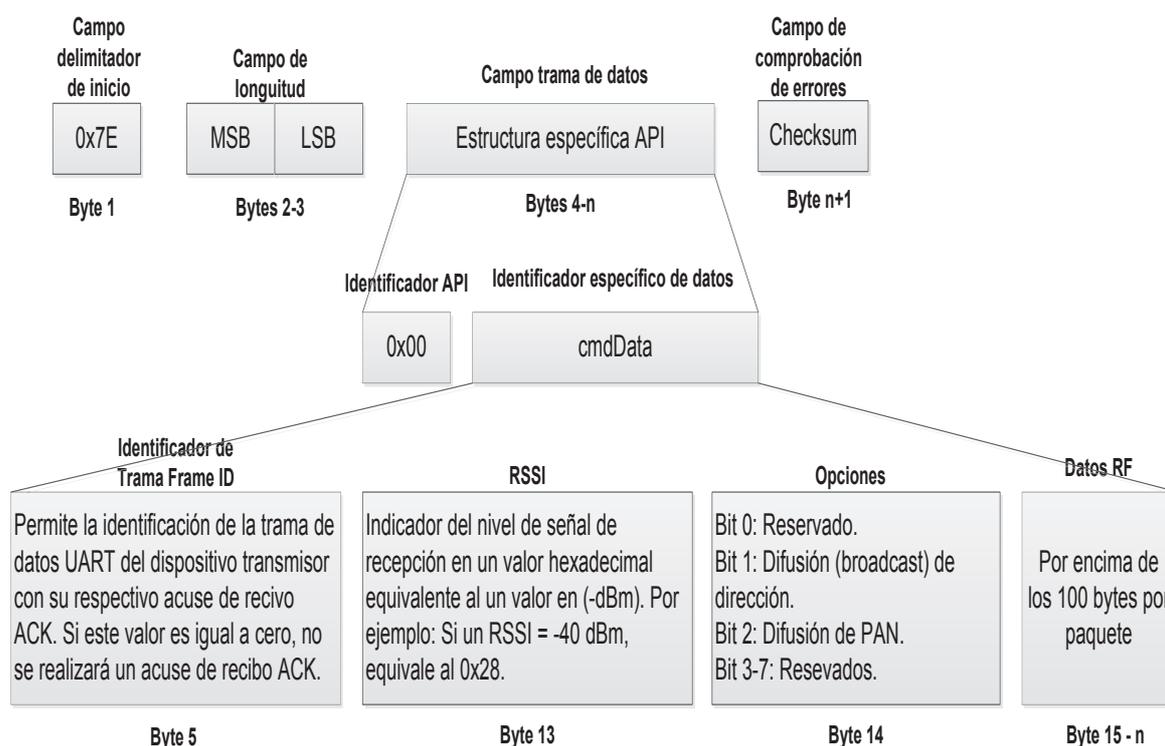
**Figura 3.57** Diagrama de Flujo de Funciones de Verificación del Campo de Datos RF y de Extracción de los Strings de Datos tanto del Módulo de Temperatura y Humedad Relativa como del Módulo de Precipitación

*Fuente: Elaboración propia*

<sup>17</sup> El Anexo C1.4 muestra el diagrama de bloques detallado de obtención del string de datos RF de la trama API de recepción.

### 3.4.3.4. Sistema de Construcción de Tramas API de Transmisión de 64-bits

La función de este sistema consiste en construir una trama de transmisión de 64-bits<sup>18</sup> que presente un identificador API igual a 0x00. Esta trama presenta una estructura tal como se presenta en la Figura 3.58. Por tanto, se procede con la construcción de los diferentes campos de la trama API de transmisión de 64-bits, tal como se muestra en la Tabla 3.21.



**Figura 3.58 Trama API de Transmisión de 64-bits de Direccionamiento.**

*Fuente: Elaboración propia basado en [28, p. 60]*

#### **Ejemplo:**

Se requiere enviar información de temperatura correspondiente a 20.5°C al módulo de temperatura y humedad relativa para presentación en el LCD, tomando en cuenta que no es necesario acuses de recibo ACK, ni control de flujo. Determinar cuál debe ser la estructura de la trama API de transmisión de 64-bits.

<sup>18</sup> El Anexo C1.5 muestra detalladamente el diagrama de bloques de la construcción de una trama API de transmisión de 64-bits de direccionamiento, la cual contiene los valores procesados de temperatura y humedad.

**Tabla 3.21 Construcción de una Tama API de Transmisión de 64-bits de Direccionamiento**

Posición	Campo	Descripción
Byte 1	Delimitador de inicio	0x7E
Byte 2	Longitud MSB	En caso de que los datos RF sean extensos
Byte 3	Longitud LSB	1byte API ID + 1byte Frame ID + 8bytes de dirección de destino + 1byte opciones + #bytes de datos RF
Byte 4	API ID	0x00, correspondiente a la trama de transmisión de 64-bits
Byte 5	Identificador de trama (Frame ID)	Un valor hexadecimal aleatorio que sea manejado por el usuario
Bytes 6-9	Dirección de destino MSB	4 bytes de longitud
Bytes 10 – 13	Dirección de destino LSB	4 bytes de longitud
Byte 14	Opciones	Permite la habilitación de ACK
Byte 15 – n	Datos RF	Txx.x_F ó Hxx.x_F, donde x representa valores numéricos de temperatura o humedad
Byte n+1	Checksum	Se obtiene de la diferencia entre 0xFF y los 8bits menos significativos obtenidos de la suma de los caracteres hexadecimales de todos los campos con excepción de los campos de delimitador de inicio y longitud.

*Fuente: Elaboración propia basado en [28]*

#### a) Delimitador de Inicio

Como ya se mencionó el delimitador de inicio es 0x7E.

#### b) Campo de Longitud.

Temperatura = 20.5 = (0x32)(0x30)(0x2E)(0x35) = 4bytes de datos RF

MSB = 0x00

LSB = 1byte API ID + 1byte Frame ID + 8bytes de dirección de destino + 1byte opciones + 4bytes de datos RF = 15bytes = 0x0F

#### c) Identificador API (API ID)

API ID = 0x00

d) Identificador de Trama (Frame ID).

Frame ID = 0x01, el cual es un valor aleatorio.

e) Dirección de Destino del Módulo de Temperatura y Humedad Relativa.

MSB = 0x0013A200; LSB = 0x406F502C

f) Opciones

No se incorpora ACK ni control de flujo = 0x01

g) Datos RF

Temperatura = 20.5 = (0x32)(0x30)(0x2E)(0x35)

h) Checksum

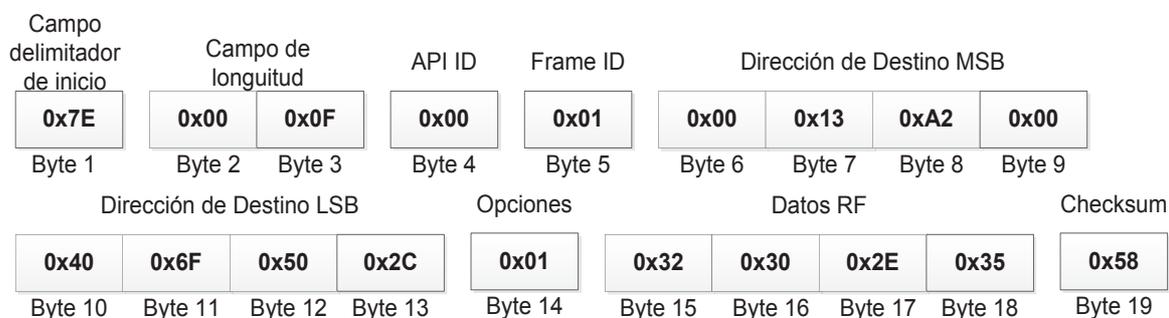
El cálculo de checksum se puede observar en la Tabla 3.22.

**Tabla 3.22 Cálculo del CheckSum para una Trama API Tx de 64-bits de Direccionamiento.**

	API ID	0x00
	Frame ID	0x01
	Dirección de destino MSB	0x00
		0x13
		0xA2
		0x00
	Dirección de destino LSB	0x40
		0x6F
		0x50
		0x2C
	Opciones	0x01
	Datos RF	0x32
		0x30
		0x2E
		0x35
	<b>TOTAL</b>	<b>0x2A7</b>

Se toma los 8 bits menos significativos de la suma total: 0xA7

Checksum = 0xFF – 0xA7 = 0x58



**Figura 3.59 Estructura de una Trama API Tx de 64-bits de Direccionamiento que Incluye Datos de Temperatura**

*Fuente: Elaboración propia*

### 3.4.4. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE ADQUISICIÓN DE DATOS DE TEMPERATURA Y HUMEDAD RELATIVA Y NIVEL DE VOLTAJE DE BATERÍA A

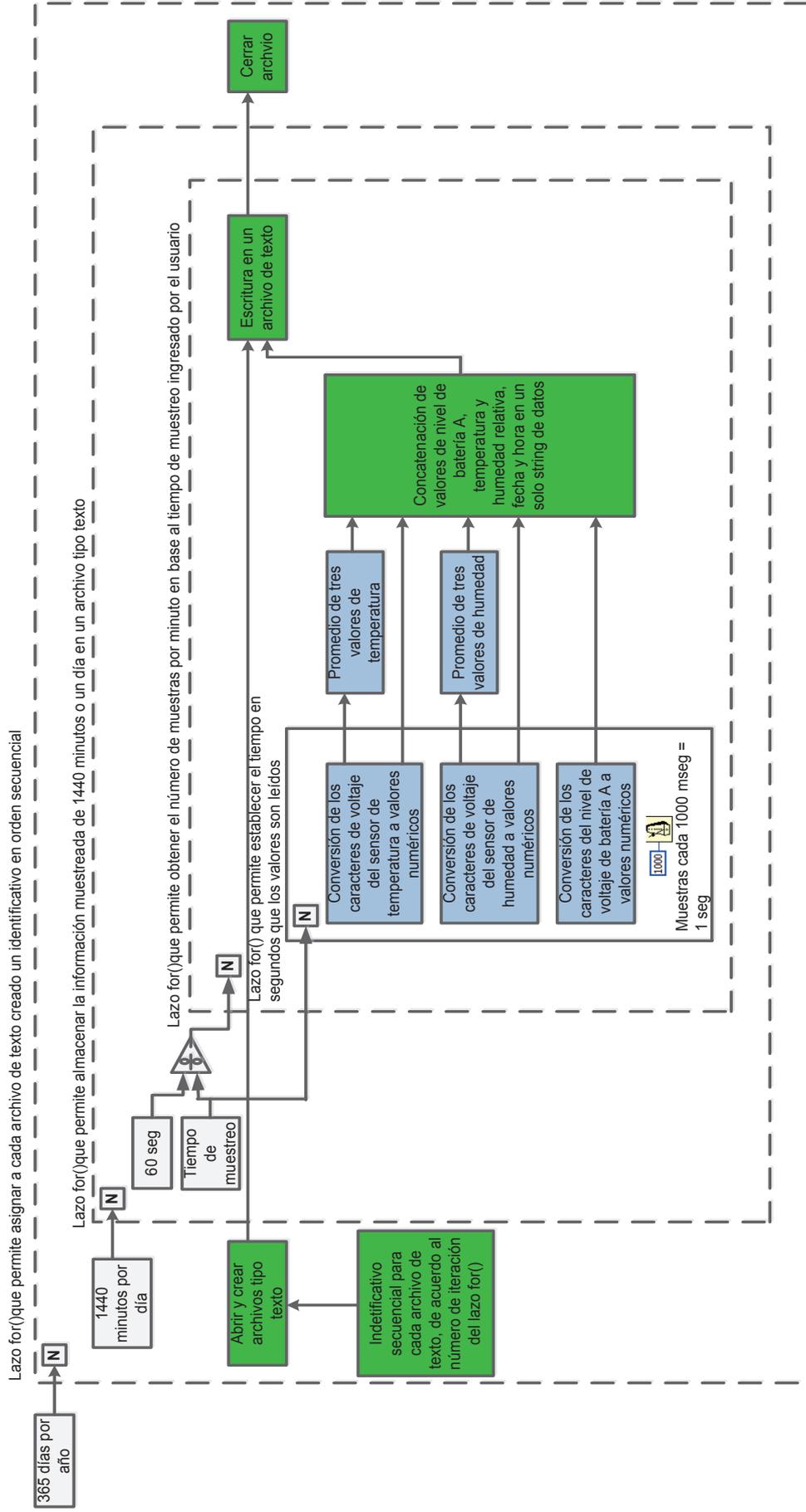
El módulo de adquisición de datos de temperatura y humedad relativa se representa en el diagrama de bloques de la Figura 3.60<sup>19</sup>, en la cual se puede observar que este módulo está conformado por dos sistemas que permiten procesar y almacenar la información del módulo de temperatura y humedad relativa.

#### 3.4.4.1. Sistema de Procesamiento y Muestreo de Información del Módulo de Temperatura y Humedad Relativa

Este sistema tiene como propósito convertir los caracteres de voltaje del sensor de temperatura y humedad obtenidos del módulo de Tx/Rx serial API, en valores numéricos de temperatura y humedad<sup>20</sup> en base a la utilización de una ecuación lineal para cada parámetro, cuyas constantes han sido encontradas de forma empírica en el laboratorio de meteorológica del INAMHI.

<sup>19</sup> En el Anexo C2 se observa el diagrama de bloques detallado de la adquisición de temperatura, humedad y nivel de batería A.

<sup>20</sup> El Anexo C2.1 y C2.2 presenta el diagrama de bloques del cálculo de los valores de temperatura y humedad relativa respectivamente.



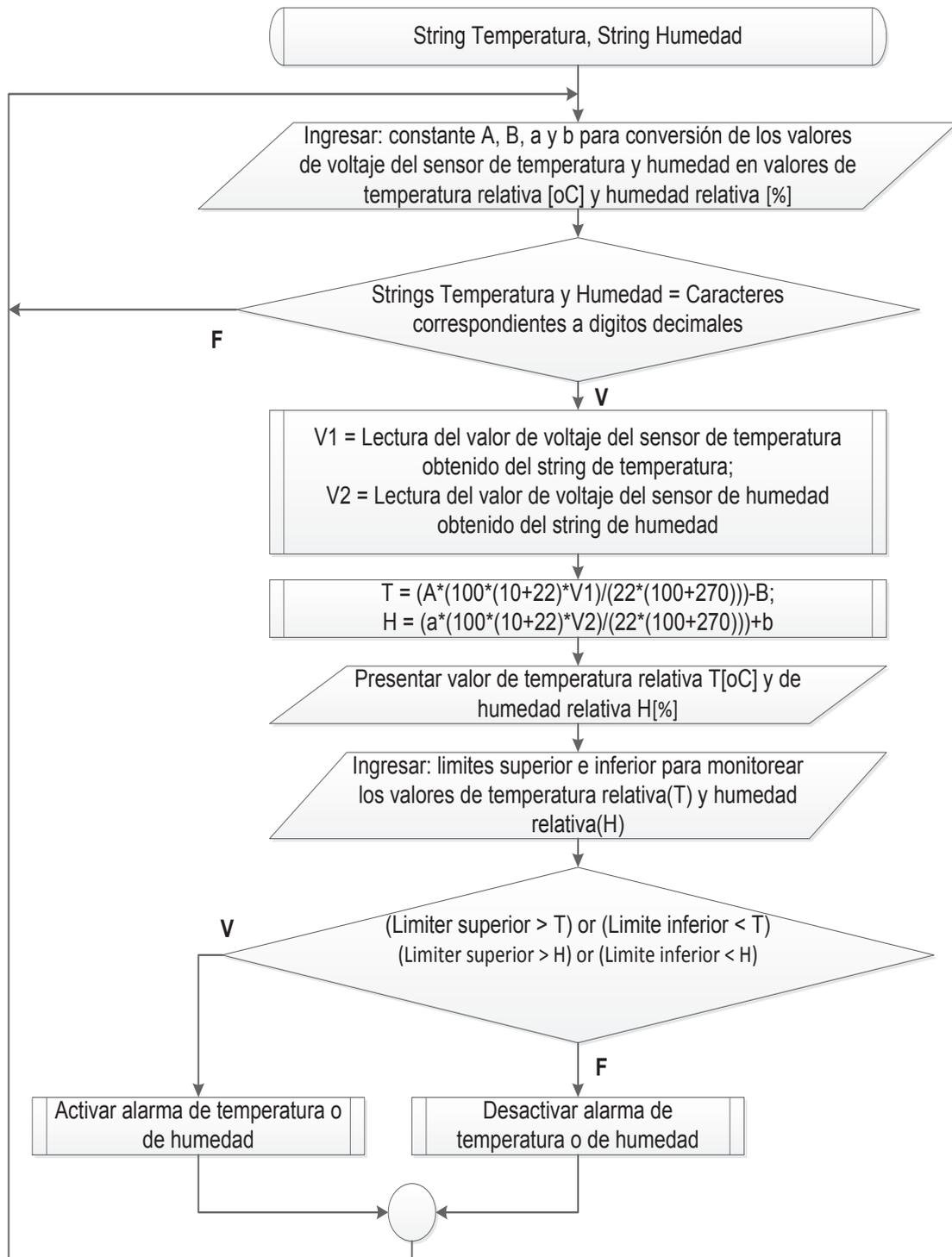
■ Sistema de Procesamiento y Muestreo de Información del Módulo de Temperatura y Humedad Relativa

■ Sistema de Almacenamiento y Creación de un Archivo de Texto de la Información Muestreada del Módulo de Temperatura y Humedad Relativa

**Figura 3.60 Diagrama de Bloques del Módulo de Adquisición de Datos de Temperatura, Humedad Relativa y Nivel de Voltaje de Batería A**

*Fuente: Elaboración propia*

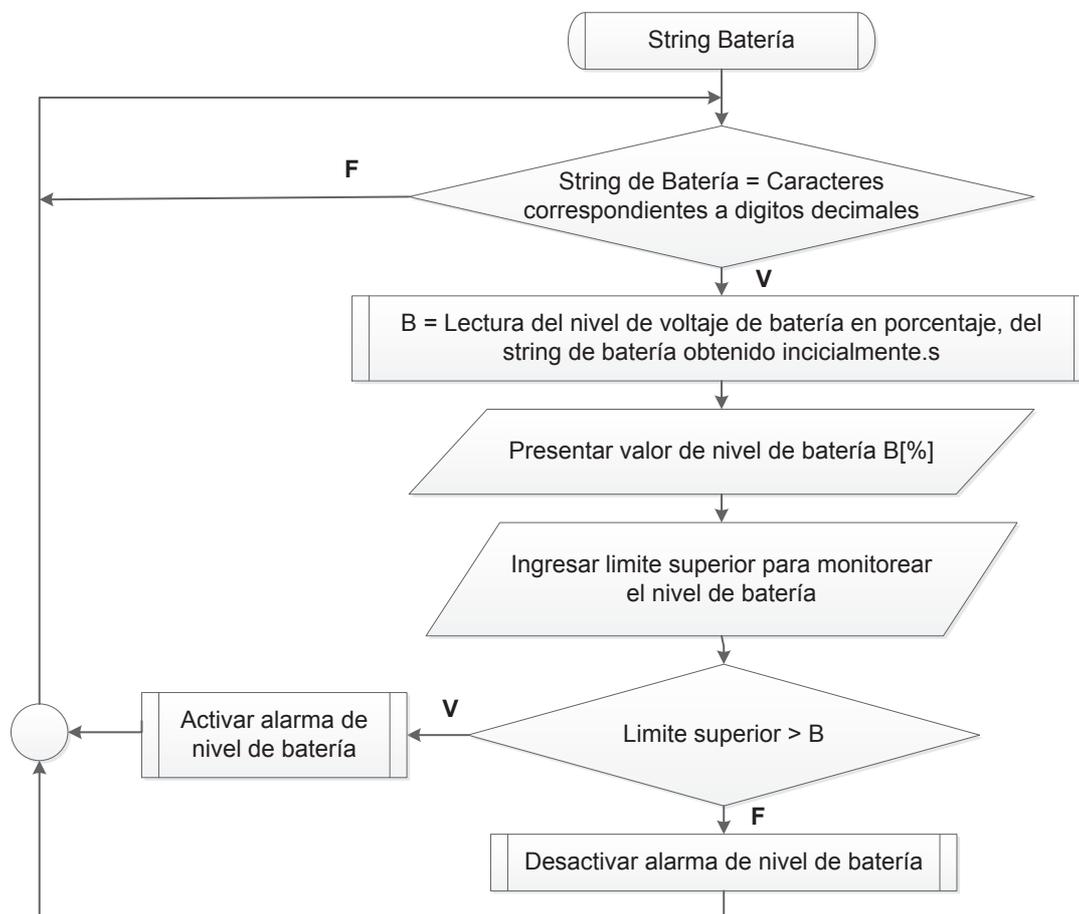
La función que cumplen con el procesamiento de las señales de voltaje del sensor de temperatura y humedad relativa se muestra en el diagrama de flujo de la Figura 3.61.



**Figura 3.61 Diagramas de Flujo de las Funciones de Procesamiento y Muestreo de Información de Temperatura y Humedad Relativa**

*Fuente: Elaboración propia*

También se implementa una función que permita obtener valores numéricos en porcentaje del nivel de voltaje necesario de la batería para funcionamiento del circuito<sup>21</sup>, tal como se muestra en la Figura 3.62.



**Figura 3.62 Diagrama de Flujo de la Función de Procesamiento de Información del Nivel de Voltaje de Batería A**

*Fuente: Elaboración propia*

De igual manera, en este sistema se establecen alarmas que se activan cuando se sobrepasan los límites superior o inferior previamente ingresados por el usuario, para los parámetros de temperatura y humedad relativa respectivamente. En el caso del porcentaje de nivel de batería, se establece una alarma que se activa cuando se excede el límite inferior ingresada por el usuario, y que a su vez indique el estado de la batería.

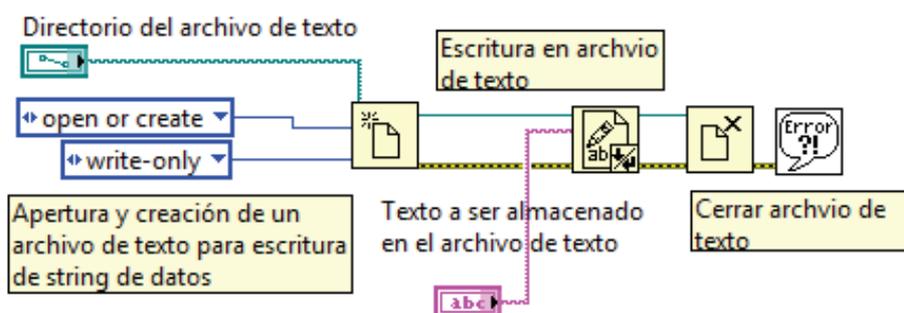
<sup>21</sup> El Anexo C2.3 muestra el diagrama de bloques detallado para obtención de los valores de nivel de batería A y B.

Finalmente, este sistema realiza promedios de tres valores de temperatura y humedad relativa previamente muestreados, de acuerdo al tiempo de muestreo ingresado por el usuario y establecidos por el INAMHI como: 20 segundos para temperatura y humedad, y 10 segundos para precipitación.

#### 3.4.4.2. Sistema de Almacenamiento y Creación de un Archivo de Texto de la Información Muestreada del Módulo de Temperatura y Humedad Relativa

Este sistema tiene como propósito concatenar y organizar por fecha y hora en una tabla de datos<sup>22</sup>, información de: temperatura relativa, temperatura promedio, humedad relativa, humedad promedio, nivel de batería A, alarma de temperatura, alarma de humedad, alarma de batería A, para su correspondiente almacenamiento en un archivo de texto y presentación.

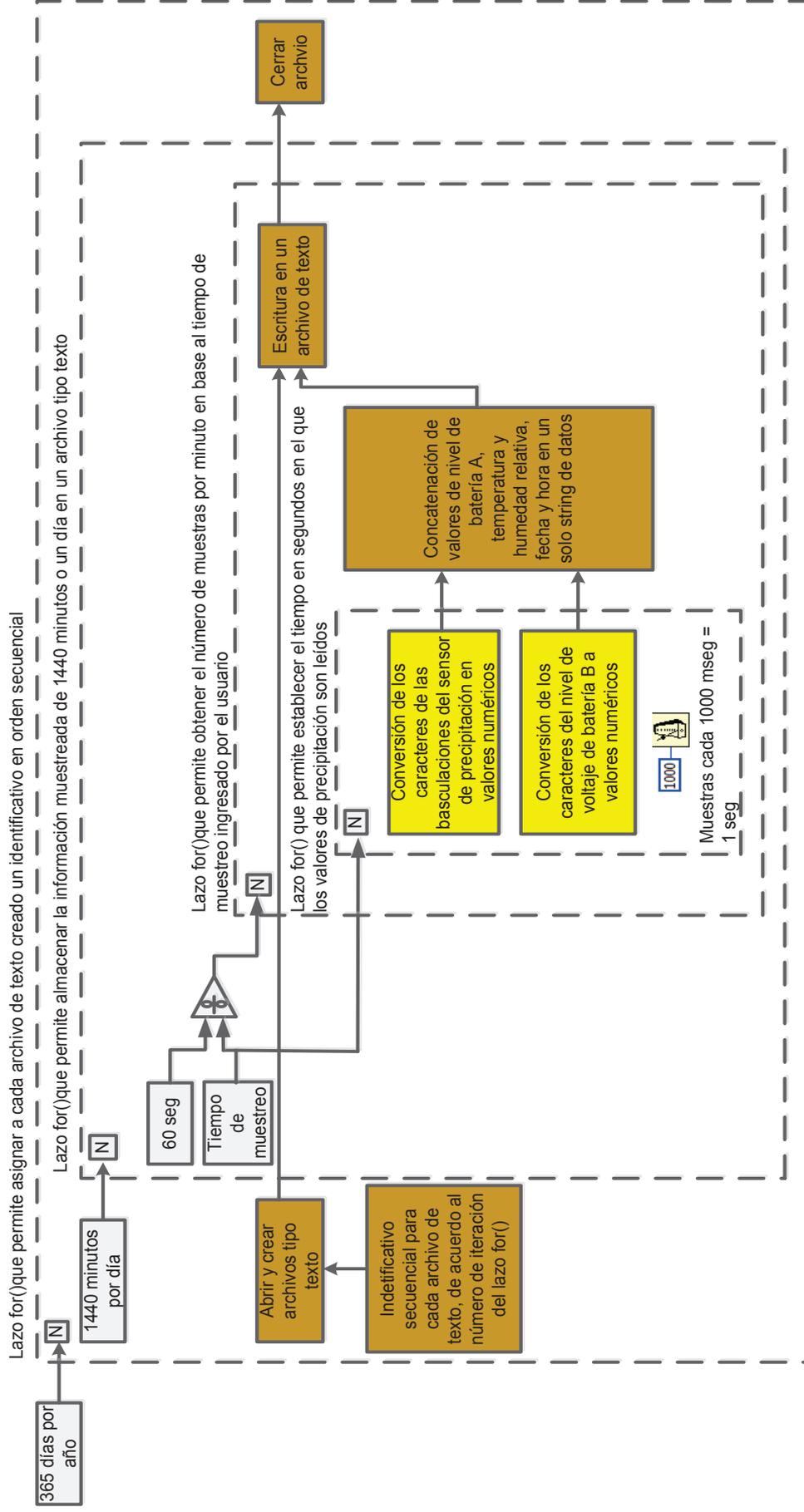
El proceso de almacenamiento de información se realiza en cada intervalo o tiempo de muestreo ingresado por el usuario, y el diagrama de bloques que lo define se presenta en la Figura 3.63. Los archivos de texto se crean cada día y contienen dicha información muestreada. Se debe mencionar, que la presentación de información a diferencia del almacenamiento, se realiza cada minuto y se puede observar en una tabla localizada en la ventana correspondiente de este módulo.



**Figura 3.63** Proceso de Almacenamiento de la Información Muestreada en un Archivo de Texto

*Fuente: Elaboración propia*

<sup>22</sup> El Anexo C2.4, muestra el diagrama de bloques detallado de la obtención de una tabla de valores de temperatura, humedad y batería A organizados por fecha y hora.



■ Sistema de Procesamiento y Muestreo de Información del Módulo de Precipitación

■ Sistema de Almacenamiento y Creación de un Archivo de Texto de la Información Muestreada del Módulo de Precipitación

■ Sistema de Bloques del Módulo de Adquisición de Datos de Precipitación y Nivel de Voltaje de Batería B

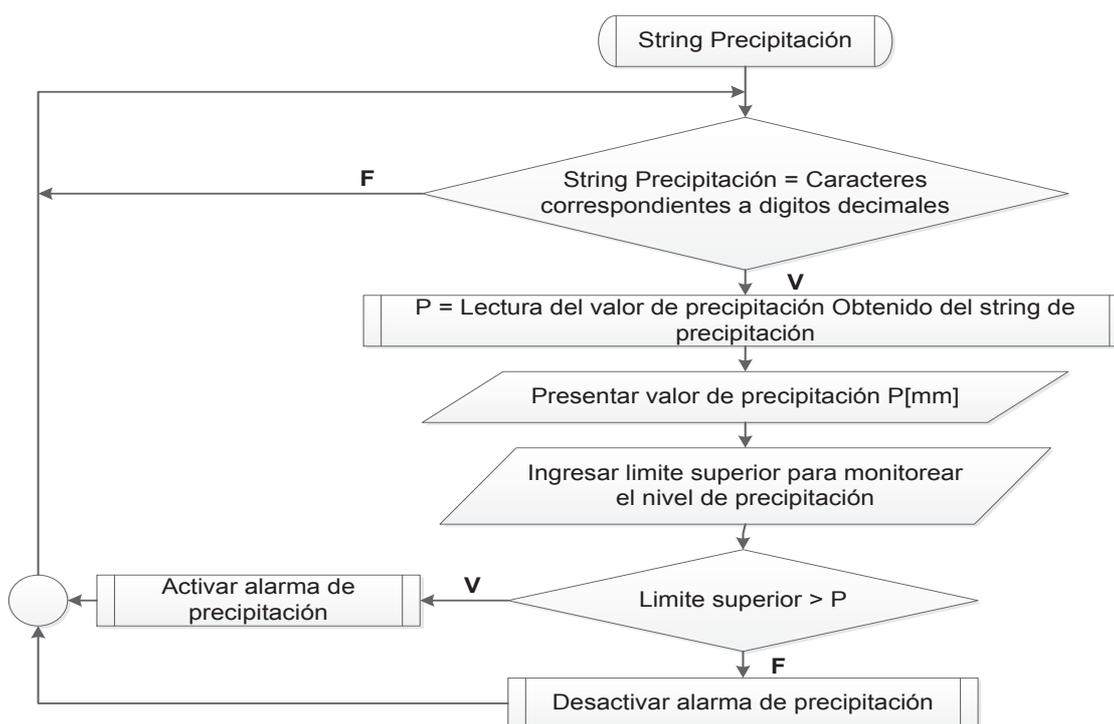
Fuente: *Elaboración propia*

### 3.4.5. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE ADQUISICIÓN DE DATOS DE PRECIPITACIÓN Y NIVEL DE VOLTAJE DE BATERÍA B

El módulo de adquisición de datos de precipitación se representa en el diagrama de bloques de la Figura 3.64<sup>23</sup>, en la cual se puede observar que el mismo está conformado por dos subsistemas que permiten procesar y almacenar la información del módulo de precipitación.

#### 3.4.5.1. Sistema de Procesamiento y Muestreo de Información del Módulo de Precipitación

Este sistema tiene como propósito convertir los caracteres de las basculaciones del sensor de precipitación<sup>24</sup> obtenidos del módulo de Tx/Rx serial API, en valores numéricos de precipitación, tal como se muestra en la Figura 3.65.



**Figura 3.65 Diagramas de Flujo de la Función de Procesamiento y Muestreo de Información de Precipitación**

*Fuente: Elaboración propia*

<sup>23</sup> El Anexo C3 muestra el diagrama de bloques detallado de la adquisición de datos de precipitación.

<sup>24</sup> El Anexo C3.1 presenta el diagrama de bloques de la obtención de los valores de precipitación.

Además, se debe tomar en cuenta que este sensor se calibró en el laboratorio de meteorología previamente. La función que permite obtener valores numéricos en porcentaje del nivel de voltaje necesario de la batería B, es la misma que se presentó para el módulo de temperatura y humedad y cuyo diagrama de flujo se representa en la Figura 3.62. De igual manera, en este sistema se establece una alarma de precipitación la cual se activa cuando se excede un límite inferior previamente ingresado por el usuario. Tal como se mencionó anteriormente, el tiempo de muestreo de precipitación establecido por el INAMHI es de 10 segundos.

#### **3.4.5.2. Sistema de Almacenamiento y Creación de un Archivo de Texto de la Información Muestreada del Módulo de Precipitación**

Este sistema tiene como propósito el de concatenar y organizar por fecha y hora en una tabla de datos<sup>25</sup>, información de: precipitación, nivel de batería B, alarma de precipitación, alarma de batería B, para su correspondiente almacenamiento en un archivo de texto y presentación. El proceso de almacenamiento y creación de un archivo de texto de información muestreada de precipitación, se realiza tal como se mostró en la sección 3.4.4.2 para el módulo de temperatura y humedad y que a su vez se observa en la Figura 3.63.

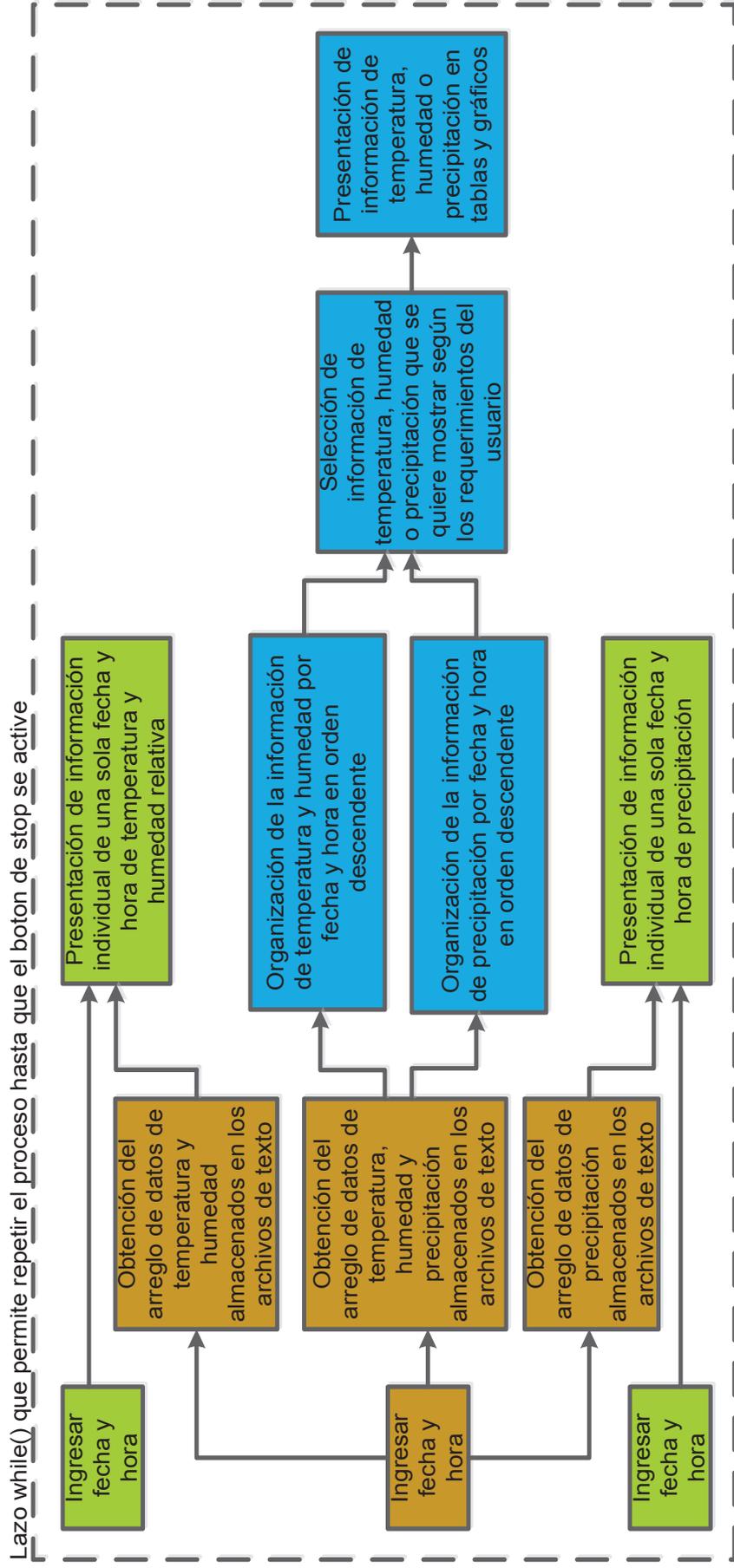
#### **3.4.6. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE PRESENTACIÓN DE DATOS METEOROLÓGICOS**

El módulo de presentación de datos meteorológicos se representa en el diagrama de bloques de la Figura 3.66<sup>26</sup>, en el cual se puede observar que el mismo está conformado por algunos sistemas que permiten organizar la información por fecha y hora, seleccionar y presentar la información de los módulos de precipitación y de temperatura y humedad relativa en conjunto y de acuerdo a los requerimientos del usuario.

---

<sup>25</sup> El Anexo C3.2 presenta el diagrama de bloques de la tabla de valores de precipitación y batería B.

<sup>26</sup> En el Anexo C4 se observa el diagrama de bloques de la presentación de datos meteorológicos.



■ Sistema de Selección de Información Meteorológica

■ Sistema de Presentación de Datos Individuales

■ Sistema de Presentación de Información de Temperatura, Humedad y Precipitación Organizada por Fecha y Hora en Orden Descendente, en Tablas y Gráficos

Figura 3.66 Diagrama de Bloques del Módulo de Presentación de Datos Meteorológicos.

Fuente: Elaboración propia.

### 3.4.6.1. Sistema de Selección de Información Meteorológica

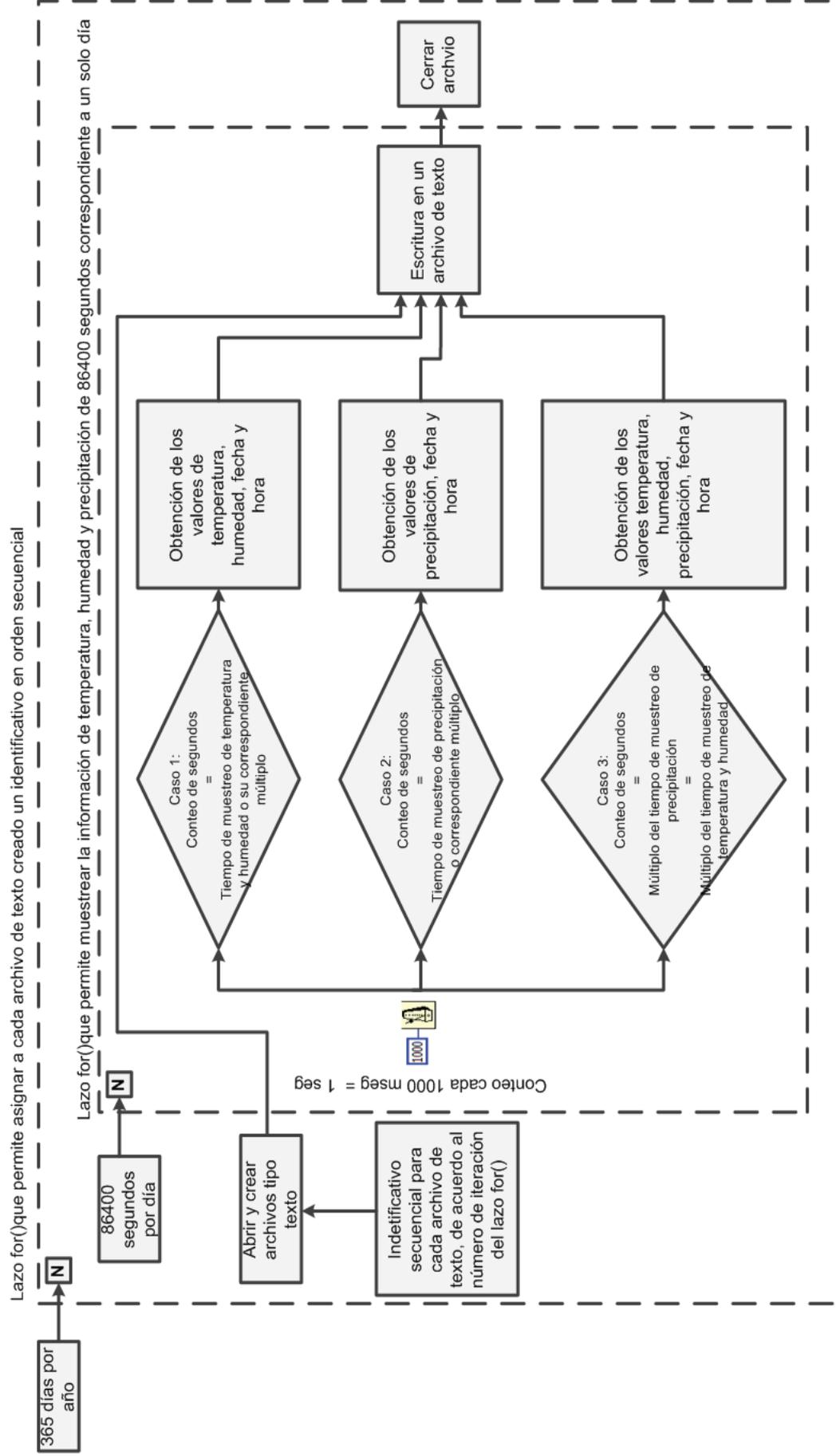
Este sistema tiene como objetivo obtener una tabla de datos de temperatura y humedad relativa, tabla de datos de precipitación o una tabla de datos de temperatura, humedad y precipitación en orden descendente por fecha y hora, en base a la acumulación de información almacenada ya sea por día, mes o año y de acuerdo a la fecha ingresada por el usuario. Cabe destacar que la información es leída de los archivos de datos tipo texto, dentro de los cuales se encuentra almacenada los datos muestreados y que han sido creados previamente en los módulos de adquisición de datos de temperatura, humedad y en el módulo de precipitación. Además, el archivo de texto que contiene los datos muestreados de temperatura, humedad y precipitación en conjunto, ordenados por fecha y hora en orden descendente, ha sido obtenido mediante la utilización de un subprograma cuyo diagrama de bloques se observa en la Figura 3.67<sup>27</sup>, y que se ejecuta en forma paralela a los demás módulos del prototipo.

El procedimiento que este sistema conlleva es el siguiente:

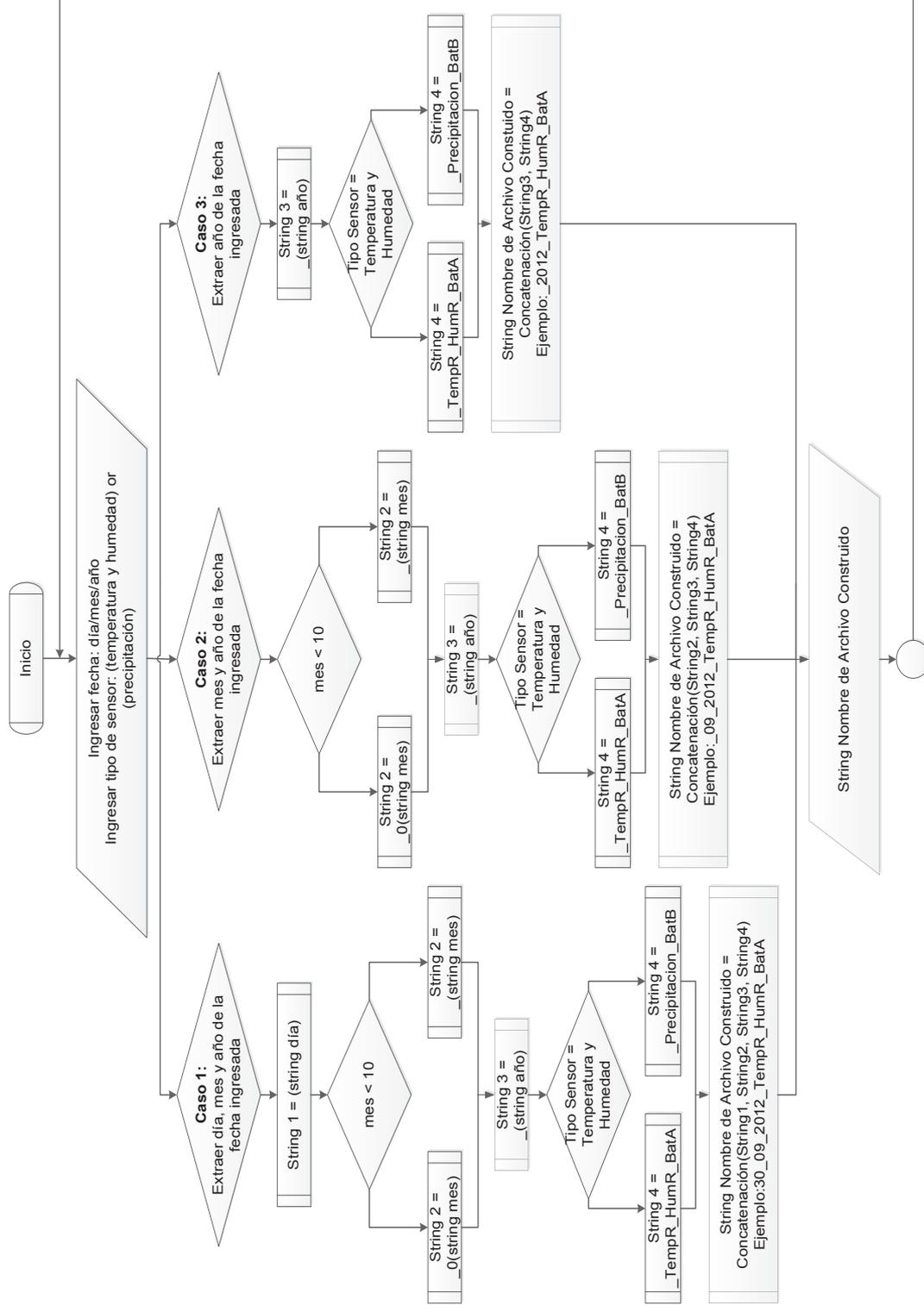
- 1) Construcción de un *string* correspondiente a la fecha en la cual el usuario quiera visualizar la información adquirida, tal como se observa en la Figura 3.68. En este sentido, si el usuario requiere observar la información correspondiente a un solo día, el *string* debe especificar día, mes y año. En el caso de que se quiera visualizar la información de todo un mes, el *string* debe especificar el mes y el año, y si se requiere observar la información correspondiente a todo el año, el *string* debe especificar el año. Es decir el *string* permitirá encontrar los archivos de texto que contienen la información, de acuerdo a la fecha ingresada por el usuario y a la cantidad de datos ya sea por día, mes o año, que se requiera observar. Los tipos de datos pueden ser solo temperatura y humedad, solo de precipitación o los tres en conjunto.

---

<sup>27</sup> El Anexo C4.1 presenta el diagrama de bloques detallado la concatenación de datos de un archivo de texto seleccionado.

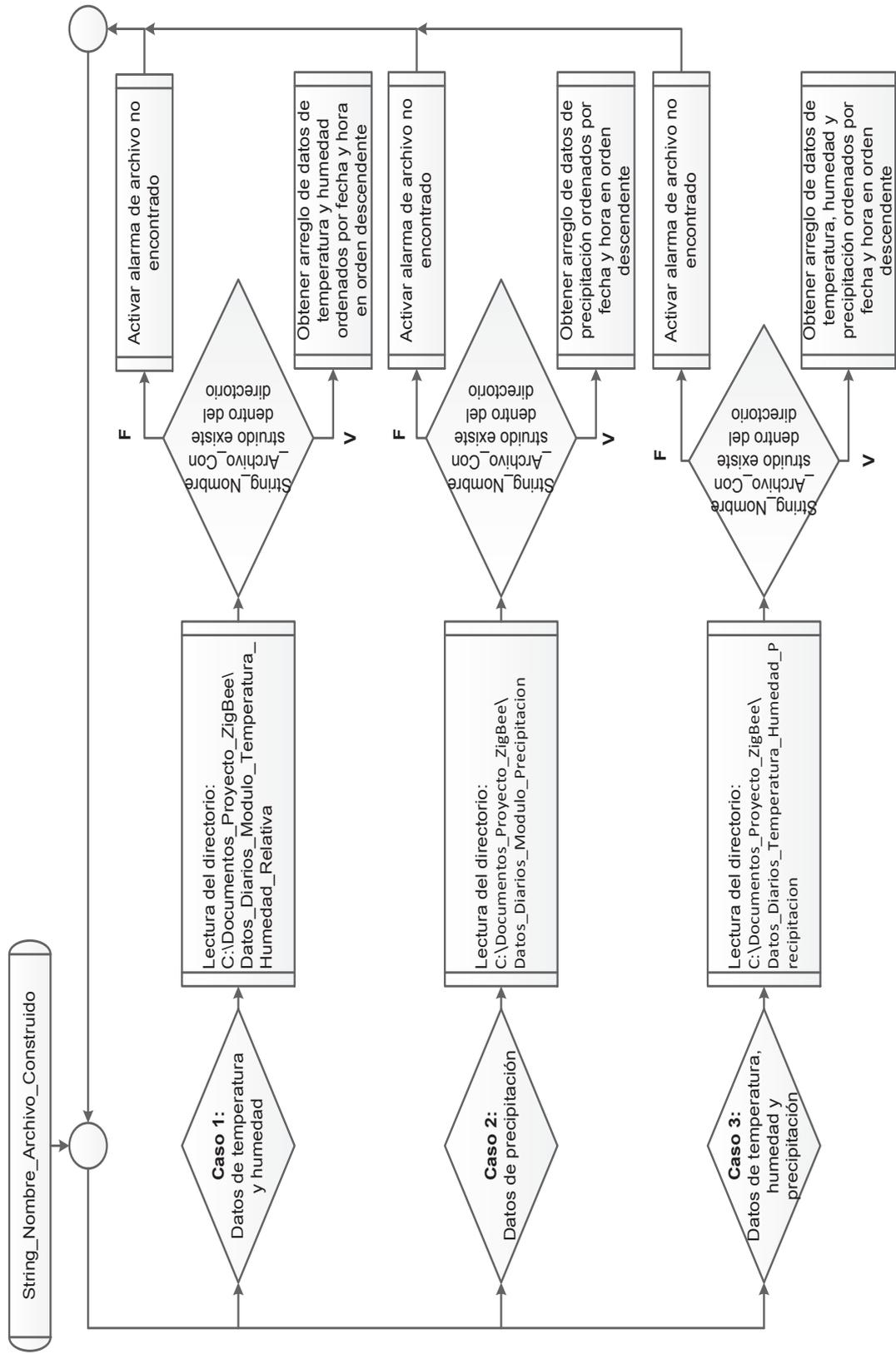


**Figura 3.67 Diagrama de Bloques del Subproma de Creación de un Archivo de Texto que contiene Información de Temperatura, Humedad y Precipitación, Ordenada y Muestreada por Fecha y Hora en Orden Descendente**  
 Fuente: *Elaboración propia*



**Figura 3.68** Diagrama de Flujo para Obtención de un String de Información que Permita Extraer los Datos Muestreados de Temperatura, Humedad o Precipitación de los Archivos de Texto en Base a la Información Ingresada por el Usuario

Fuente: *Elaboración propia*



**Figura 3.69** Diagrama de Flujo para Organización de un Arreglo de Directorios que Indiquen la Dirección de los Archivos de Texto de los cuales se Extraen los Datos Muestreados en Arreglos de Datos

*Fuente: Elaboración propia*

- 2) En base al string construido previamente para encontrar archivos de texto que contengan la información que se requiere visualizar, se determina y organiza un arreglo de directorios o paths que indiquen donde se encuentran almacenados los archivos de texto, de los cuales se extrae la información en arreglos de temperatura y humedad, precipitación, o los tres parámetros en conjunto tal como observa en el diagrama de flujo de la Figura 3.69. La Figura 3.70, muestra los resultados de esta función.

Arreglo de Datos de TR y HR										Arreglo de Datos de Precipitación					
12/12/2012	19:01:12	14.1	4.7	39.7	13.2	0	Normal	Normal	Alarma	12/12/2012	19:01:02	0.1	100	Normal	Normal
12/12/2012	19:01:32	14.1	9.4	39.7	26.5	0	Normal	Normal	Alarma	12/12/2012	19:01:12	0.2	100	Normal	Normal
12/12/2012	19:01:52	14.6	14.2	39.4	39.6	0	Normal	Normal	Alarma	12/12/2012	19:01:22	0.4	100	Normal	Normal
12/12/2012	19:02:12	14.6	4.9	40.2	13.4	0	Normal	Normal	Alarma	12/12/2012	19:01:32	0.4	100	Normal	Normal
12/12/2012	19:02:32	14.1	9.5	39.4	26.5	0	Normal	Normal	Alarma	12/12/2012	19:01:42	0.4	100	Normal	Normal
12/12/2012	19:02:52	14.1	14.2	39.4	39.7	0	Normal	Normal	Alarma	12/12/2012	19:01:52	0.0	100	Normal	Normal
12/12/2012	19:03:12	14.1	4.7	40.2	13.4	0	Normal	Normal	Alarma	12/12/2012	19:02:02	0.0	100	Normal	Normal
12/12/2012	19:03:32	14.1	9.4	39.7	26.6	0	Normal	Normal	Alarma	12/12/2012	19:02:12	0.0	100	Normal	Normal
12/12/2012	19:03:52	14.6	14.2	40.2	40.0	0	Normal	Normal	Normal	12/12/2012	19:02:22	0.0	100	Normal	Normal
12/12/2012	19:04:12	14.1	4.7	39.4	13.1	0	Normal	Normal	Normal	12/12/2012	19:02:32	0.0	100	Normal	Normal
12/12/2012	19:04:32	14.1	9.4	39.4	26.3	0	Normal	Normal	Normal	12/12/2012	19:02:42	0.0	100	Normal	Normal
12/12/2012	19:04:52	13.8	14.0	39.7	39.5	0	Normal	Normal	Normal	12/12/2012	19:02:52	0.0	100	Normal	Normal
12/12/2012	19:05:12	14.1	4.7	40.2	13.4	0	Normal	Normal	Normal	12/12/2012	19:03:02	0.0	100	Normal	Normal
12/12/2012	19:05:32	14.1	9.4	39.4	26.5	0	Normal	Normal	Normal	12/12/2012	19:03:12	0.0	100	Normal	Normal
12/12/2012	19:05:52	15.6	14.6	42.2	40.6	0	Normal	Normal	Normal	12/12/2012	19:03:22	0.0	100	Normal	Normal
12/12/2012	19:06:12	15.6	5.2	42.2	14.1	0	Normal	Normal	Normal	12/12/2012	19:03:32	0.0	100	Normal	Normal
12/12/2012	19:06:32	15.6	10.4	42.8	28.3	0	Normal	Normal	Normal	12/12/2012	19:03:42	0.0	100	Normal	Normal
12/12/2012	19:06:52	15.6	15.6	42.2	42.4	0	Normal	Normal	Normal	12/12/2012	19:03:52	0.0	100	Normal	Normal
12/12/2012	19:07:12	15.6	5.2	42.2	14.1	0	Normal	Normal	Normal	12/12/2012	19:04:02	0.0	100	Normal	Normal

**Figura 3.70 Arreglos de Datos Organizado por Fecha y Hora de: (a) Temperatura y Humedad (b) Precipitación**

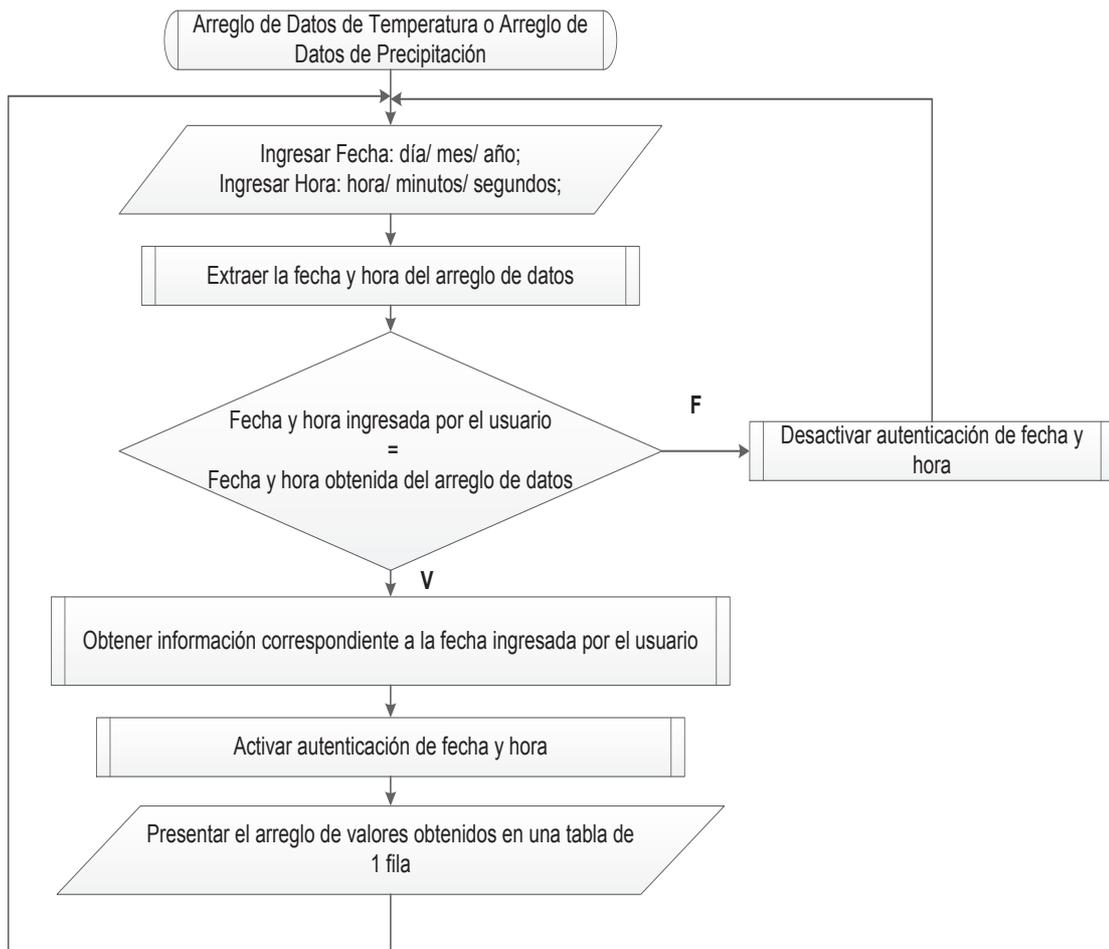
*Fuente: Elaboración propia utilizando el software LabView 2010*

### 3.4.6.2. Sistema de Presentación de Datos Individuales

Este sistema permite presentar la información individual, ya sea del módulo de temperatura y humedad relativa o del módulo de precipitación, de una sola fecha ingresada por el usuario. La función del diagrama de la Figura 3.71<sup>28</sup>, tiene como objetivo obtener los valores de temperatura, temperatura promedio, humedad,

<sup>28</sup> El Anexo C4.2 muestra el diagrama de bloques detallado de la obtención de información meteorológica correspondiente a una sola fecha específica.

humedad promedio, nivel de batería A, estado del sensor HMP45A y estado batería A, en una sola tabla de una fila, tal como se muestra en la Figura 3.72(a). Mientras tanto, los valores de precipitación, estado sensor de precipitación, nivel de batería B y estado batería B, se presentan en una tabla de una fila como se muestra en la Figura 3.72(b)



**Figura 3.71 Diagrama de Flujo para Presentar la Información Individual de una Sola Fecha Ingresada por el Usuario**

Fuente: Elaboración propia

Temperatura Relativa [oC]	Temperatura Promedio [oC]	Humedad Relativa [%]	Humedad Promedio [%]	Nivel Batería A [%]	Estado Sensor
14.1	4.7	39.7	13.2	0	Normal

(a)

Precipitación [mm]	Nivel Batería B	Estado Sensor Precipitación	Estado Nivel Batería B
0.2	100	Normal	Normal

(b)

**Figura 3.72 Presentación de la Información Individual de una Sola Fecha Ingresada por el Usuario de: (a) Temperatura y Humedad Relativa (b) Precipitación**

Fuente: Elaboración propia utilizando el software LabView 2010

### 3.4.6.3. Sistema de Presentación de Información de Temperatura, Humedad y Precipitación Organizada por Fecha y Hora en Orden Descendente, en Tablas y Gráficos

Este sistema tiene como propósito el de presentar la información de temperatura, humedad y precipitación, en tablas y gráficos para su correspondiente observación en una ventana de interfaz con el usuario. En este sentido, el proceso que conlleva al desarrollo de este sistema es:

- 1) Obtener los gráficos de temperatura vs tiempo y humedad vs tiempo<sup>29</sup>, a partir del arreglo de datos de temperatura y humedad relativa obtenido previamente en la sección 1.1.1.1.
- 2) Obtener el gráfico de precipitación vs tiempo<sup>30</sup>, a partir del arreglo de datos de precipitación obtenido previamente en la sección 1.1.1.1.
- 3) Filtrar la información de temperatura, humedad o precipitación que se quiere mostrar de acuerdo a los requerimientos del usuario, tal como se observa en la Tabla 3.23, donde 0L oculta la información de la correspondiente columna y 1L muestra la información de la correspondiente columna.

**Tabla 3.23 Presentación u Ocultación de Información de Temperatura, Humedad y Precipitación según Requerimientos del Usuario**

Precipitación[mm] Gráfica: Precipitación[mm] vs Tiempo	Humedad[%] Gráfica: Humedad[%] vs Tiempo	Temperatura[°C] Gráfica: Temperatura[°C] vs Tiempo
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

*Fuente: Elaboración propia*

<sup>29</sup> El Anexo C4.3 presenta el diagrama de bloques que permite obtener las gráficas y arreglo de datos de temperatura y humedad relativa.

<sup>30</sup> El Anexo C4.4 muestra el diagrama de bloques para obtener la gráfica y arreglo de datos de precipitación.

### 3.4.7. VENTANA DE INFORMACIÓN METEOROLÓGICA

La ventana de información meteorológica tiene como fin el de facilitar al usuario la gestión y manejo de la información procesada por el módulo de adquisición de datos, para lo cual, se debe integrar en un solo programa todos los módulos desarrollados y diseñados en secciones anteriores, de tal forma que se ejecuten de forma simultánea. Las diferentes opciones que presenta esta ventana son:

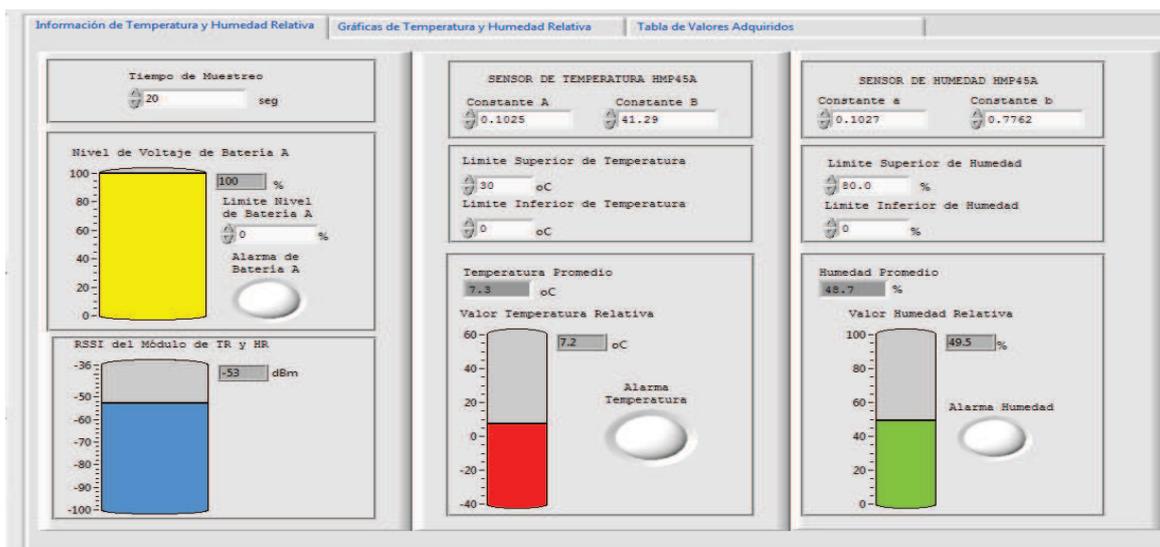
- a) Inicio.- Para presentación de la información general del módulo de adquisición de datos, tal como se muestra en la Figura 3.73.



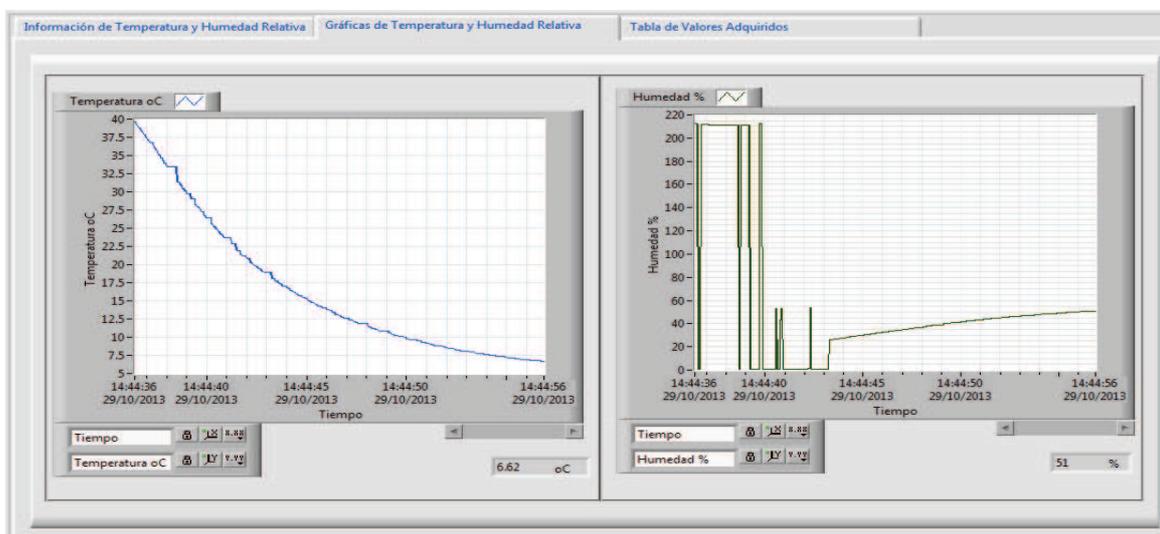
**Figura 3.73** Ventana de Inicio para Presentación de Información General del Módulo de Adquisición de Datos

*Fuente: Elaboración propia*

- b) Información de Temperatura y Humedad.- Para observación de las pestañas que permiten gestionar los datos obtenidos del módulo de temperatura y humedad relativa, tal como se muestra en la Figura 3.74.



(a)



(b)

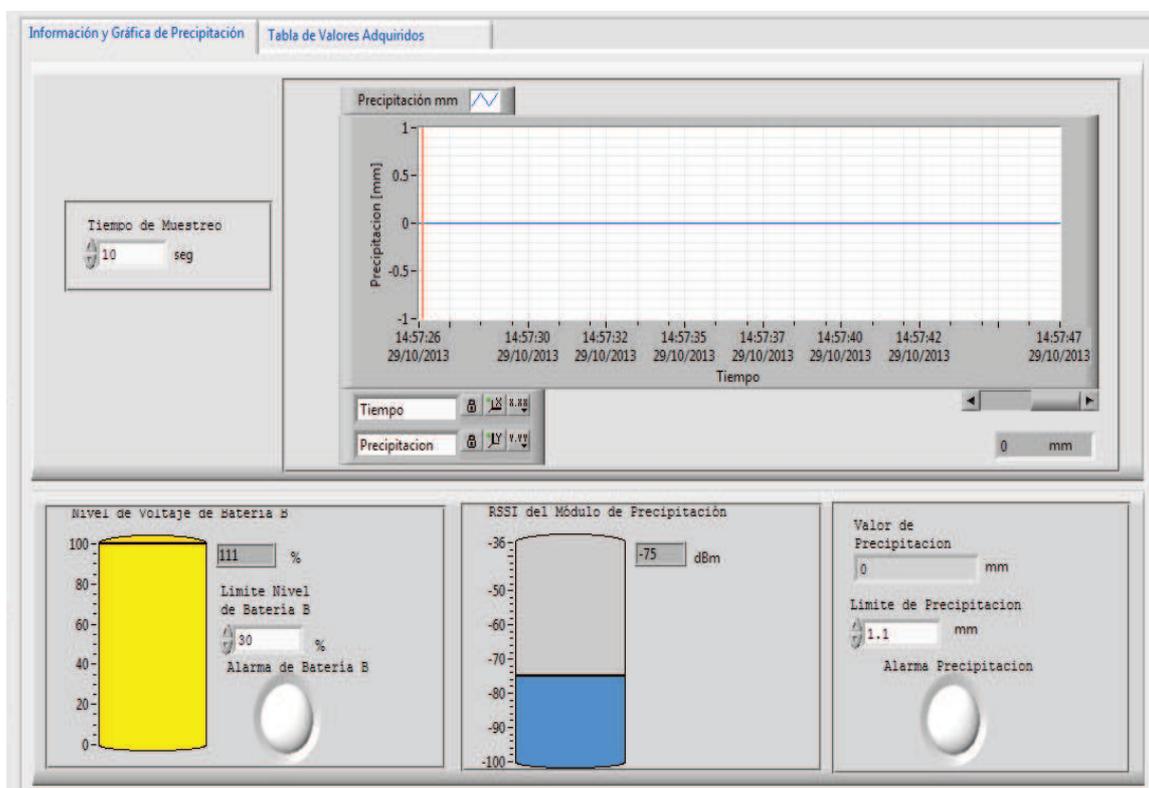
Fecha	Hora	Temperatura Relativa [oC]	Temperatura Promedio [oC]	Humedad Relativa [%]	Humedad Promedio [%]	Nivel Bate
29-octubre-2013	11:42:52	20.3	20.3	81.0	81.1	100
29-octubre-2013	11:43:52	19.9	20.1	73.8	76.4	100
29-octubre-2013	11:44:52	18.5	18.9	54.4	58.6	100
29-octubre-2013	11:45:52	17.1	17.4	53.9	53.6	100
29-octubre-2013	11:46:52	16.2	16.3	56.7	56.3	100
29-octubre-2013	11:47:52	14.6	15.0	61.2	60.3	100
29-octubre-2013	11:48:52	13.9	14.2	63.8	62.6	100
29-octubre-2013	11:49:52	13.3	13.5	65.6	65.1	100
29-octubre-2013	11:50:52	12.6	12.8	67.8	67.1	100
29-octubre-2013	11:51:52	12.1	12.4	69.4	68.6	100
29-octubre-2013	11:52:52	11.7	11.9	70.9	70.4	100
29-octubre-2013	11:53:52	11.4	11.5	72.0	71.5	100
29-octubre-2013	11:54:52	11.1	11.3	73.0	72.6	100
29-octubre-2013	11:55:52	10.9	11.0	74.0	73.6	100
29-octubre-2013	11:56:52	10.8	10.8	74.7	74.4	100
29-octubre-2013	11:57:52	10.6	10.6	75.4	75.2	100
29-octubre-2013	11:58:52	10.4	10.4	75.9	75.8	100
29-octubre-2013	11:59:52	10.3	10.3	76.4	76.3	100
29-octubre-2013	12:00:52	10.2	10.2	77.1	76.9	100
29-octubre-2013	12:01:52	10.1	10.1	77.4	77.2	100
29-octubre-2013	12:02:52	10.0	10.0	77.6	77.6	100
29-octubre-2013	12:03:52	10.0	10.0	78.1	77.9	100
29-octubre-2013	12:04:52	9.9	9.9	78.2	78.2	100
29-octubre-2013	12:05:52	9.9	9.8	78.5	78.5	100
29-octubre-2013	12:06:52	9.8	9.8	78.7	78.6	100
29-octubre-2013	12:07:52	9.8	9.8	79.0	78.9	100
29-octubre-2013	12:08:52	9.7	9.7	79.1	79.1	100
29-octubre-2013	12:09:52	9.7	9.7	79.2	79.2	100
29-octubre-2013	12:10:52	9.7	9.7	79.4	79.3	100

(c)

Figura 3.74 Pestañas del Módulo de Adquisición de Datos de Temperatura y Humedad: (a) Información de Temperatura y Humedad Relativa, (b) Gráficas de Temperatura y Humedad Relativa y (c) Tabla de Valores Adquiridos

Fuente: Elaboración propia

- c) Información de Precipitación.- Para observación de las pestañas que permiten gestionar los datos obtenidos del módulo de precipitación, tal como se muestra en la Figura 3.75.



(a)

Figure 3.75(b) displays the 'Tabla de Valores Adquiridos' (Acquired Values Table) interface. It shows a table with the following columns: Fecha, Hora, Precipitación [mm], Nivel de Batería B, Estado Precipitación, and Estado Nivel de Batería B. The table contains 24 rows of data, all showing 0.0 mm of precipitation and 111% battery level.

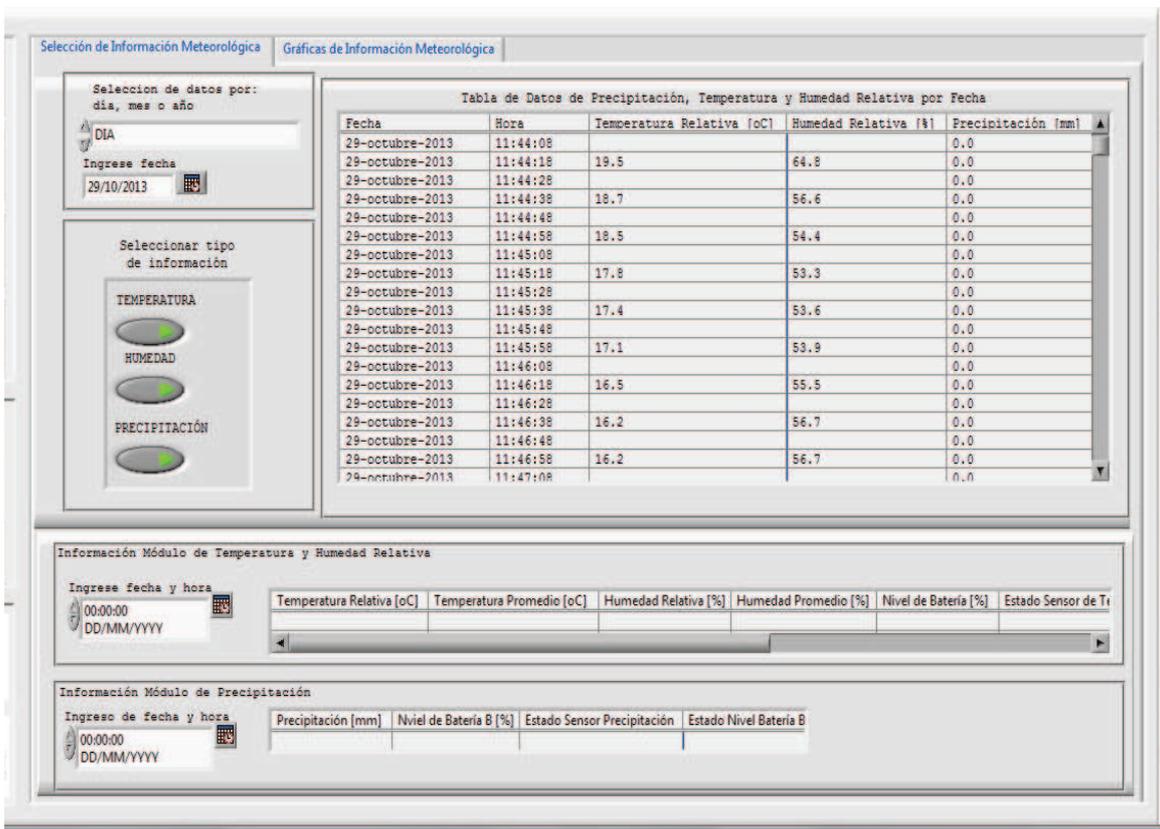
Fecha	Hora	Precipitación [mm]	Nivel de Batería B	Estado Precipitación	Estado Nivel de Batería B
29-octubre-2013	11:42:52	0.0	111	Normal	Normal
29-octubre-2013	11:43:52	0.0	111	Normal	Normal
29-octubre-2013	11:44:52	0.0	111	Normal	Normal
29-octubre-2013	11:45:52	0.0	111	Normal	Normal
29-octubre-2013	11:46:52	0.0	111	Normal	Normal
29-octubre-2013	11:47:52	0.0	111	Normal	Normal
29-octubre-2013	11:48:52	0.0	111	Normal	Normal
29-octubre-2013	11:49:52	0.0	111	Normal	Normal
29-octubre-2013	11:50:52	0.0	111	Normal	Normal
29-octubre-2013	11:51:52	0.0	111	Normal	Normal
29-octubre-2013	11:52:52	0.0	111	Normal	Normal
29-octubre-2013	11:53:52	0.0	111	Normal	Normal
29-octubre-2013	11:54:52	0.0	111	Normal	Normal
29-octubre-2013	11:55:52	0.0	111	Normal	Normal
29-octubre-2013	11:56:52	0.0	111	Normal	Normal
29-octubre-2013	11:57:52	0.0	111	Normal	Normal
29-octubre-2013	11:58:52	0.0	111	Normal	Normal
29-octubre-2013	11:59:52	0.0	111	Normal	Normal
29-octubre-2013	12:00:52	0.0	111	Normal	Normal
29-octubre-2013	12:01:52	0.0	111	Normal	Normal
29-octubre-2013	12:02:52	0.0	111	Normal	Normal
29-octubre-2013	12:03:52	0.0	111	Normal	Normal
29-octubre-2013	12:04:52	0.0	111	Normal	Normal
29-octubre-2013	12:05:52	0.0	111	Normal	Normal
29-octubre-2013	12:06:52	0.0	111	Normal	Normal
29-octubre-2013	12:07:52	0.0	111	Normal	Normal
29-octubre-2013	12:08:52	0.0	111	Normal	Normal
29-octubre-2013	12:09:52	0.0	111	Normal	Normal

(b)

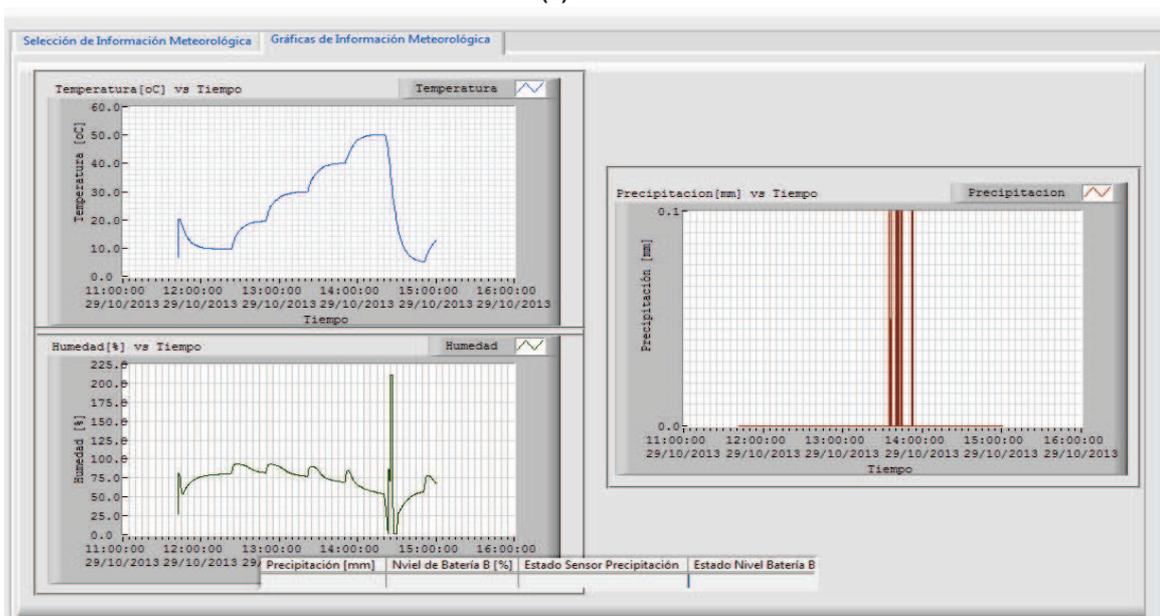
Figura 3.75 Pestañas del Módulo de Adquisición de Datos de Precipitación: (a) Información y Gráfica de Precipitación, (b) Tabla de Valores Adquiridos

Fuente: Elaboración propia

d) Presentación de Información Meteorológica.- Para observación de la recopilación de información tanto de temperatura, humedad y precipitación, tal como se muestra en la Figura 3.76.



(a)



(b)

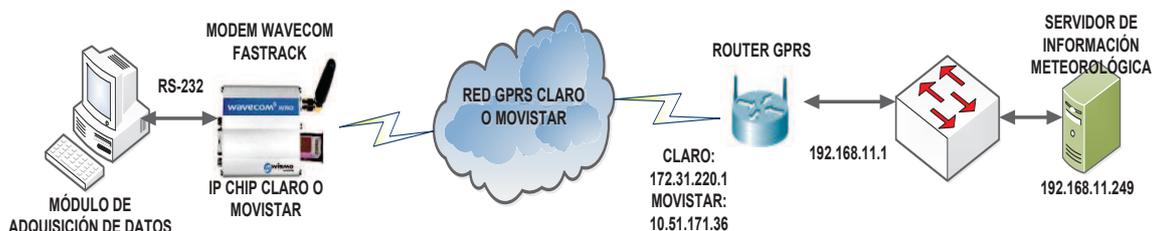
Figura 3.76 Pestañas del Módulo de Presentación de los Datos Meteorológicos: (a) Selección de Información Meteorológica, (b) Gráficas de Información Meteorológica

Fuente: Elaboración propia

- e) STOP.- Para detener y cerrar todo el programa.
- f) VISA COM.- Para escoger el puerto de comunicación serial en el cual se conecta el dispositivo coordinador ZigBee.
- g) GPRS COM.- Para escoger el puerto de comunicación serial en el cual se conecta el modem GPRS Wavecom.

### 3.5. DISEÑO DEL MÓDULO DE TRANSMISIÓN Y PROCESAMIENTO DE DATOS GPRS [38]

La recopilación de información meteorológica por parte de la central del INAMHI localizada en Quito, se realiza por medio de la red GPRS de Claro o Movistar, para lo cual es necesario la utilización de una solución que permitan la conexión punto a punto. La solución utilizada por el INAMHI es la M2M (*Machine to Machine*), la cual permite establecer una conexión punto a punto entre un dispositivos M2M, que en nuestro caso es el modem Tx/Rx Wavecom Fastrack y el servidor de información meteorológica a través del router GPRS marca Multitech, el cual ya se encuentra configurado y en funcionamiento. La Figura 3.77 muestra la solución M2M utilizando el protocolo TCP-IP.



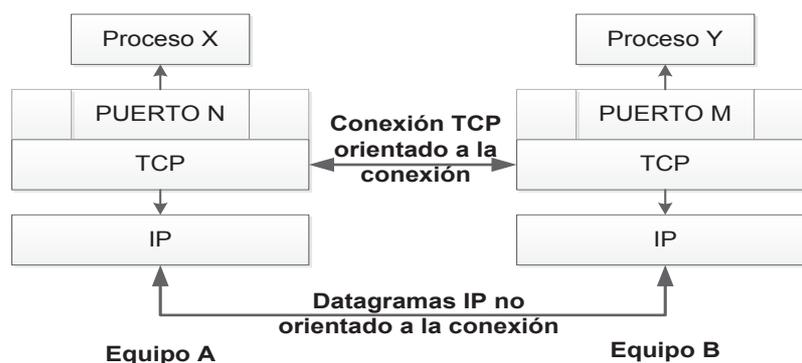
**Figura 3.77 Solución M2M para Conexión Remota de Dos Maquinas Punto a Punto Utilizando el Protocolo TCP-IP**

*Fuente: Elaboración propia*

También se debe mencionar que para diseñar el módulo de transmisión y procesamiento GPRS, se procedió con el desarrollo de programas en LabView implementados tanto en el módulo de adquisición de datos remoto como en el servidor de información meteorológica, que permiten definir procedimientos para establecer la conexión punto a punto, tomando en cuenta que la comunicación se realiza utilizando el protocolo TCP/IP sobre GPRS. En este sentido, el mecanismo de comunicación entre procesos se establece mediante la utilización de un

modelo cliente-servidor, en el cual se define *sockets*, que permiten la comunicación y presentan los siguientes atributos:

- Dominio.- Especifica el medio de comunicación de la red que el socket utilizará.
- Protocolo.- Especifica que protocolo se va a usar.
- Tipo.- Los protocolos de internet proveen dos niveles distintos de servicio: flujo y datagramas, tal como se muestra en la Figura 3.78.



**Figura 3.78 Protocolo TCP Orientado a la Conexión y Datagramas IP no Orientado a la Conexión**  
Fuente: Elaboración propia basada en [38]

En el caso de TCP/IP se utiliza *sockets* orientados a conexión, que permiten una comunicación fiable mediante el establecimiento de un circuito virtual como se observa en la Figura 3.79, para lo cual se define una dirección IP y un número de puerto tanto en el servidor como en el cliente, los cuales se definen como:

- Servidor.- Proceso o programa que ofrece un servicio y que permanece pasivo a la espera de una solicitud de conexión.
- Cliente.- Proceso o programa que envía una petición a un servidor y espera una respuesta, para la correspondiente lectura o escritura de información en el canal de comunicación previamente establecido.

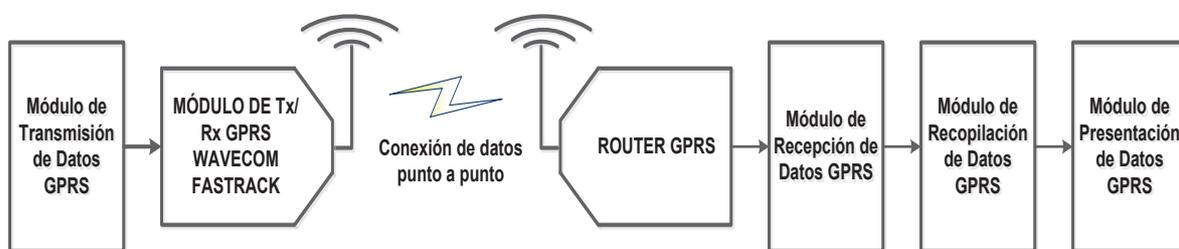


**Figura 3.79 Modelo Cliente-Servidor**  
Fuente: Elaboración propia basado en [38]

### 3.5.1. SISTEMAS Y ELEMENTOS DEL MÓDULO DE TRANSMISIÓN Y PROCESAMIENTO DE DATOS GPRS

Los módulos y elementos que se establecen para la transmisión y procesamiento GPRS se muestran en la Figura 3.80 y se enumeran como:

- a) Módulo de Transmisión GPRS.
- b) Módulo de Tx/Rx GPRS Wavecom Fastrack.
- c) Módulo de Recepción de Datos GPRS.
- d) Módulo de Recopilación de Datos GPRS.
- e) Módulo de Presentación de Datos GPRS.



**Figura 3.80** Sistemas y Elementos del Módulo de Transmisión y Procesamiento de Datos GPRS

*Fuente: Elaboración propia*

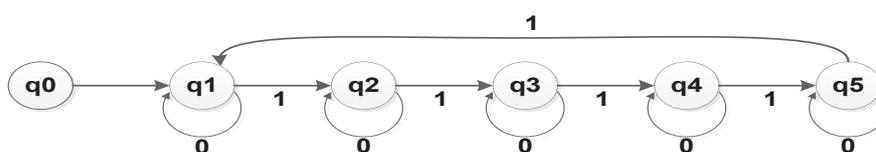
Se debe mencionar que el router inalámbrico GPRS, no se encuentra entre los elementos de transmisión o procesamiento de datos GPRS, ya que es un dispositivo que ya se encuentra configurado y en funcionamiento en el departamento IDT del INAMHI.

### 3.5.2. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE TRANSMISIÓN DE DATOS GPRS

El diseño del módulo de transmisión de datos GPRS, se realiza mediante la implementación de una máquina de estados cuyo diagrama se presenta en la Figura 3.81<sup>31</sup>. Su objetivo, es el de funcionar como servidor dentro del modelo cliente-servidor de una conexión TCP/IP, y enviar la información de temperatura,

<sup>31</sup> Los Anexos C5, C5.1, C5.2, C5.3, C5.4, C5.6 y C5.7, presentan los diagramas de bloques de los estados que permiten que la transmisión GPRS sea realizada.

humedad y precipitación procesada por el módulo de adquisición de datos, vía comunicación RS-232 al modem GPRS, para lo cual es necesario la implementación de un procedimiento que permita al modem autenticarse dentro de una red GPRS, para a continuación realizar la comunicación TCP/IP. El procedimiento para configuración del modem se realiza mediante la utilización de comandos AT, los cuales serán descritos en cada uno de los estados de la Tabla 3.24. Además, se debe señalar que la comunicación GPRS se realiza tal como se estipula en la sección 1.3.4.



**Figura 3.81** Diagrama de Estados para el Diseño del Módulo de Transmisión de Datos GPRS

*Fuente: Elaboración propia*

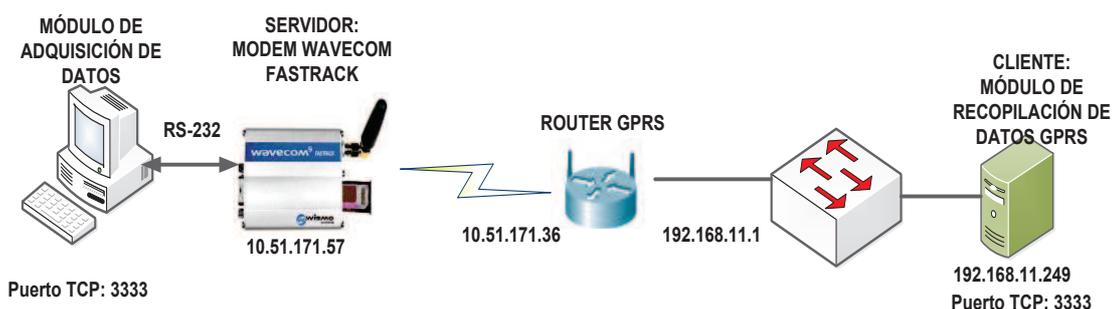
**Tabla 3.24** Sintaxis, Descripción y Respuesta de cada uno de los Comandos AT Utilizados para Iniciar la Comunicación GPRS y que Describen a cada uno de los Estados de la Transmisión GPRS

Estado Actual	Sintaxis	Respuesta y Descripción	
		1(V)	0(F)
q0	AT+CFUN = 1	OK Resetear el Modem	
q1	AT+CREG?	+CREG: 0,1 La estación móvil ha sido registrada en la red GPRS	+CREG: 2 La estación móvil no se ha registrada en la red GPRS, y sigue buscando un operador
q2	AT+CGACT?	+CGACT: 1,1 OK El contexto PDP se encuentra activado	+CGACT: 0 OK El contexto PDP se encuentra desactivado
q3	AT+CGDATA=1	CONNECT La estación móvil se encuentra autenticada y asociada al APN de la red GPRS de CLARO	ERROR La comunicación entre la estación móvil y la red GPRS ha fallado
q4	-----	Enviar la información de los valores de temperatura, humedad y precipitación con el siguiente formato: TxxxHxxxPxxxF OK	Espera a la recepción del carácter R del cliente que indique que la información ha sido recibida
q5	AT	El envío de datos ha finalizado y el canal de comunicación se ha liberado luego de 5 minutos	El envío de datos continúa y el canal de comunicación se encuentra abierto hasta recibir una respuesta del modem WAVECOM durante 5 minutos

*Fuente: Elaboración propia basado en [39]*

### 3.5.3. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE TRANSMISIÓN Y RECEPCIÓN WAVECOM FASTRACK

El módem de transmisión GPRS WAVECOM Fastrack localizado en el módulo de adquisición de datos remoto, debe ser programado de tal forma que permita la obtención de información de temperatura, humedad y precipitación en el puerto serial RS-232, para a continuación enviar dicha información vía GPRS al servidor de información meteorológica utilizando el protocolo TCP/IP. En este sentido, al módem se le asigna las funciones de un servidor y al módulo de recopilación de datos GPRS se le asigna las funciones de un cliente dentro del modelo cliente-servidor de una conexión TCP/IP, para el manejo de *sockets* previamente descrito. Se debe mencionar que la dirección IP y el puerto TCP tanto para el cliente como para el servidor se observan en la Figura 3.82, tomando en cuenta que el INAMHI proporcionó un Chip de la operadora Movistar.



**Figura 3.82 Modelo Cliente-Servidor TCP/IP y Definición de Direcciones IP y Sockets**

*Fuente: Elaboración propia*

#### 3.5.3.1. Estructura del Programa de Comunicación GPRS TCP/IP

El módem GPRS WAVECOM es programado por medio del lenguaje ANSI C, haciendo uso del software OPEN AT. Los eventos que deben ser tomados en cuenta para la correspondiente configuración y programación son:

- Identificación y autenticación del módem dentro de la red GPRS, por medio del SIM correspondiente al proveedor MOVISTAR.
- Comunicación RS-232 y configuración del *socket* correspondiente, para enviar datos del puerto serial al puerto TCP o en caso de recibir datos en este puerto, enviarlos directamente al puerto serial.

- Establecimiento de la conexión, transferencia de datos y liberación del canal de comunicación bajo el contexto del protocolo TCP/IP orientado a la conexión.

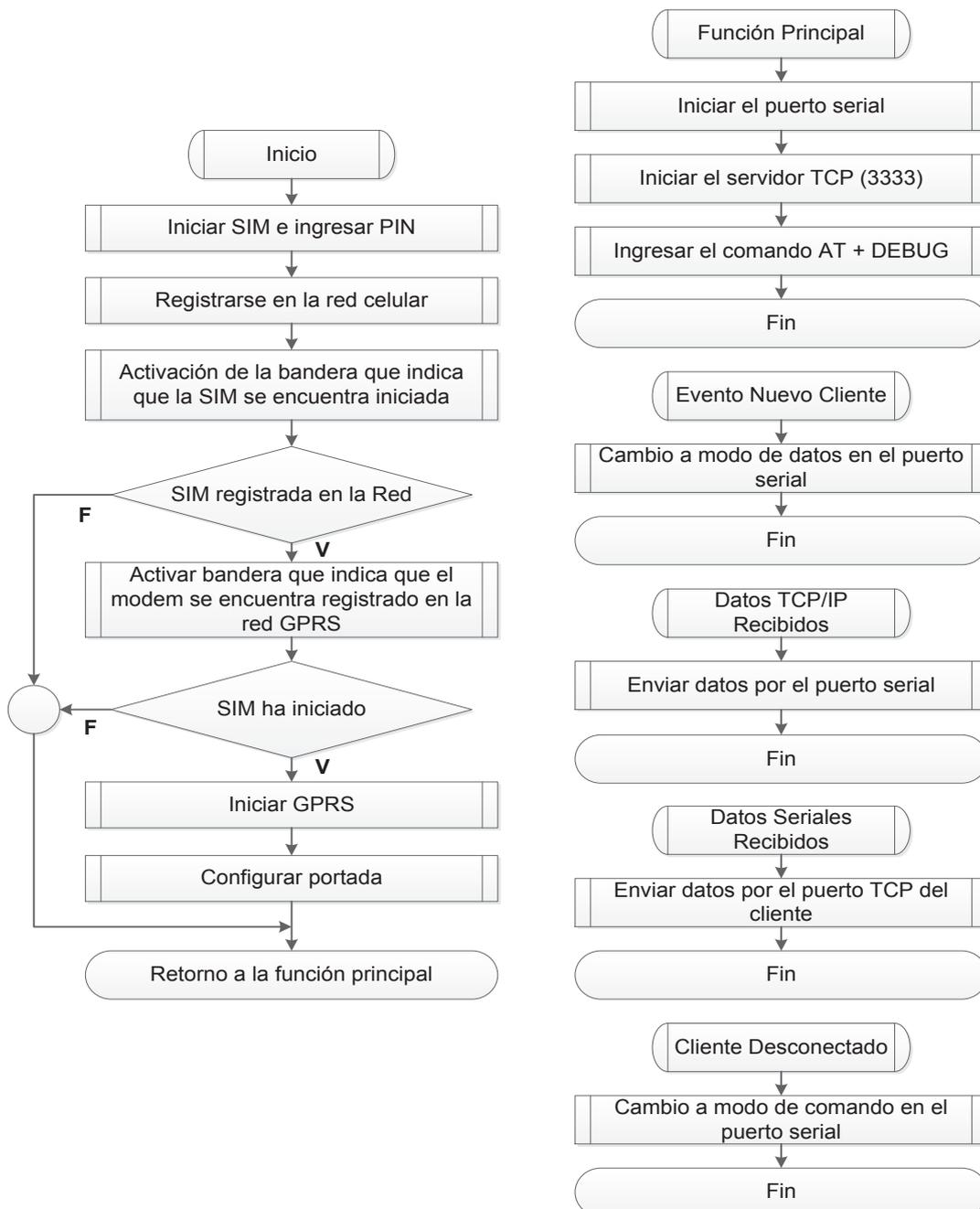
Por ende, se procede a la descripción del programa InamhiV4.dwl, tomando en cuenta la siguiente secuencia de eventos descrito en la Figura 3.83 y en el manual de ingeniería del modem WAVECOM desarrollado y provisto por el departamento IDT del INAMHI:

- 1) Llamada a la función de inicio de la tarjeta SIM
- 2) Llamada a la función de monitoreo de registro en la RED Celular
- 3) Esperar a que los eventos de SIM INICIADA y REGISTRO EXITOSO EN LA RED se cumplan.
- 4) Llamada a la función para iniciar GPRS.
- 5) Esperar a que el evento de GPRS INICIADO se cumpla.
- 6) Llamada a la función de inicio del puerto serial.
- 7) Llamada a la función de inicio del servidor TCP/IP.
- 8) Esperar a que un cliente se conecte.
- 9) Esperar a que el cliente TCP envíe datos o lo haga el puerto serial.
- 10) Si se recibe datos del cliente TCP se envía estos datos por el puerto serial.
- 11) Si se recibe datos del puerto serial se envía estos datos por el puerto del cliente TCP.
- 12) Esperar a que el evento de cliente desconectado se produzca.
- 13) Poner al puerto serial en modo de comandos.

### **3.5.3.2. Configuración del Modem WAVECOM**

La configuración del modem WAVECOM es necesaria para cumplir con los siguientes parámetros:

- Cargar el programa de comunicación GPRS TCP/IP.
- Indicar el correspondiente APN en el cual se debe asociar el modem.



**Figura 3.83** Secuencia de Eventos para Autenticación y Registro del Modem dentro de la Red GPRS.  
Fuente: [40].

De esta forma se procede a establecer la secuencia de pasos señalados en el manual de operación del modem WAVECOM provisto por el IDT del INAMHI, que permiten configurar los parámetros previamente mencionados.

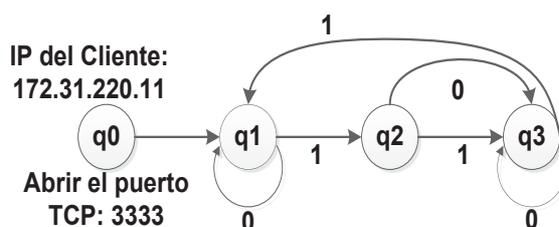
- 1) Ejecutar un programa terminal (p.e. Hyperterminal) que permita la correspondiente configuración del modem tomando en cuenta los siguientes parámetros:

- Velocidad del interfaz serial: 9600 bps
  - Bits de datos: 8 bits.
  - Paridad: Ninguna.
  - Bits de parada: 1 bit.
  - Control de flujo: Ninguno.
- 2) Verificación de la comunicación RS-232 con el modem por medio del comando AT, que el caso de ser exitosa la palabra OK aparecerá en pantalla.
  - 3) Activación de la visualización de caracteres escritos por medio del comando ATE1.
  - 4) Detener la ejecución de cualquier programa almacenado en el modem por medio del comando AT+WOPEN = 0.
  - 5) Borrar cualquier programa que se encuentre almacenado en el modem por medio del comando AT+WOPEN = 4.
  - 6) Cargar el archivo InamhiV4.dwl en el modem por medio del comando AT+WDWL, de tal forma que a continuación de la ejecución de este comando se realice la transferencia del archivo por medio de la opción 1K Xmodem del hyperterminal.
  - 7) Ejecutar el programa InamhiV4.dwl por medio del comando AT+WOPEN = 1.
  - 8) Reiniciar el modem por medio del comando AT+CFUN = 1.
  - 9) Definir la dirección IP del SIM y el correspondiente APN de la red GPRS MOVISTAR a la cual se va enganchar el modem por medio del comando AT+TIEMPO = 60,"10.51.171.36","svaestat.movistar.com.ec".
  - 10) Guardar las configuraciones realizadas por medio del comando AT&W.
  - 11) Finalmente reiniciar nuevamente el modem para ejecutar el programa desde su inicio por medio del comando AT+CFUN = 1.

Se debe reiterar que tanto la estructuración del programa InamhiV4.dwl mencionada en la sección 3.5.3.1, como la secuencia de pasos para configuración del modem WAVECOM, son provistas y desarrolladas por el departamento IDT del INAMHI.

### 3.5.4. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE RECEPCIÓN DE DATOS GPRS

El diseño del módulo de recepción de datos GPRS, se realiza mediante la implementación de una máquina de estados cuyo diagrama se presenta en la Figura 3.84<sup>32</sup>. Su objetivo, es el de funcionar como cliente dentro del modelo cliente-servidor de una conexión TCP/IP, y recibir la información de temperatura, humedad y precipitación procesada y enviada por el modem WAVECOM por medio de la red GPRS. De esta forma se procede con la implementación de un procedimiento que permita la lectura de datos en el puerto TCP 3333 para a continuación procesar dicha información en el módulo de recopilación de datos GPRS. Este procedimiento se encuentra descrito en cada uno de los estados de la Tabla 3.25.



**Figura 3.84 Diagrama de Estados para el Diseño del Módulo de Recepción de Datos GPRS**

*Fuente: Elaboración propia*

**Tabla 3.25 Función y Descripción de cada uno de los Estados de la Recepción GPRS TCP/IP**

Estado Actual	Función	Descripción	
		1(V)	0(F)
q1	Lectura de datos en el puerto TCP 3333	La lectura del carácter F indica que ha recibido la información requerida	Lectura de caracteres hasta recibir el carácter F
q2	Escritura de datos en el puerto TCP 3333	Escritura del carácter 'r' en el puerto TCP para indicar que se ha recibido la información de temperatura, humedad y precipitación al servidor	
q3	Lectura del puerto TCP 3333	Lectura del comando AT el cual indica que el servidor se encuentra a la espera de enviar información después de 5 minutos	Lectura de datos hasta recibir el comando AT del servidor

*Fuente: Elaboración propia*

<sup>32</sup> Los Anexos C6 y C6.1, presentan los diagramas de bloques de los estados que se deben ejecutar para cumplir con la recepción de datos vía GPRS.

### **3.5.5. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE RECOPIACIÓN DE DATOS GPRS**

El módulo de recopilación de datos GPRS, permite la clasificación y almacenamiento de los valores de temperatura, humedad y precipitación obtenidos por el módulo de recepción de datos GPRS, en archivos de texto separados y organizados por fecha y hora tanto para temperatura y humedad como para precipitación. El muestreo de la información de temperatura, humedad y precipitación se realiza cada 5 minutos, por otro lado, los archivos de texto son creados cada día, lo que equivale a 288 muestras cada 5 minutos por archivo. Un identificador individual por archivo revela si se trata de información de temperatura y humedad o de precipitación y a su vez, se indica en este identificador la hora, mes, día y año en el cual el archivo fue creado. De esta forma el fichero que almacena los archivos de texto se encuentra organizado en orden secuencial y cronológico. El diagrama de bloques que indica el diseño de este módulo se muestra en la Figura 3.85<sup>33</sup>.

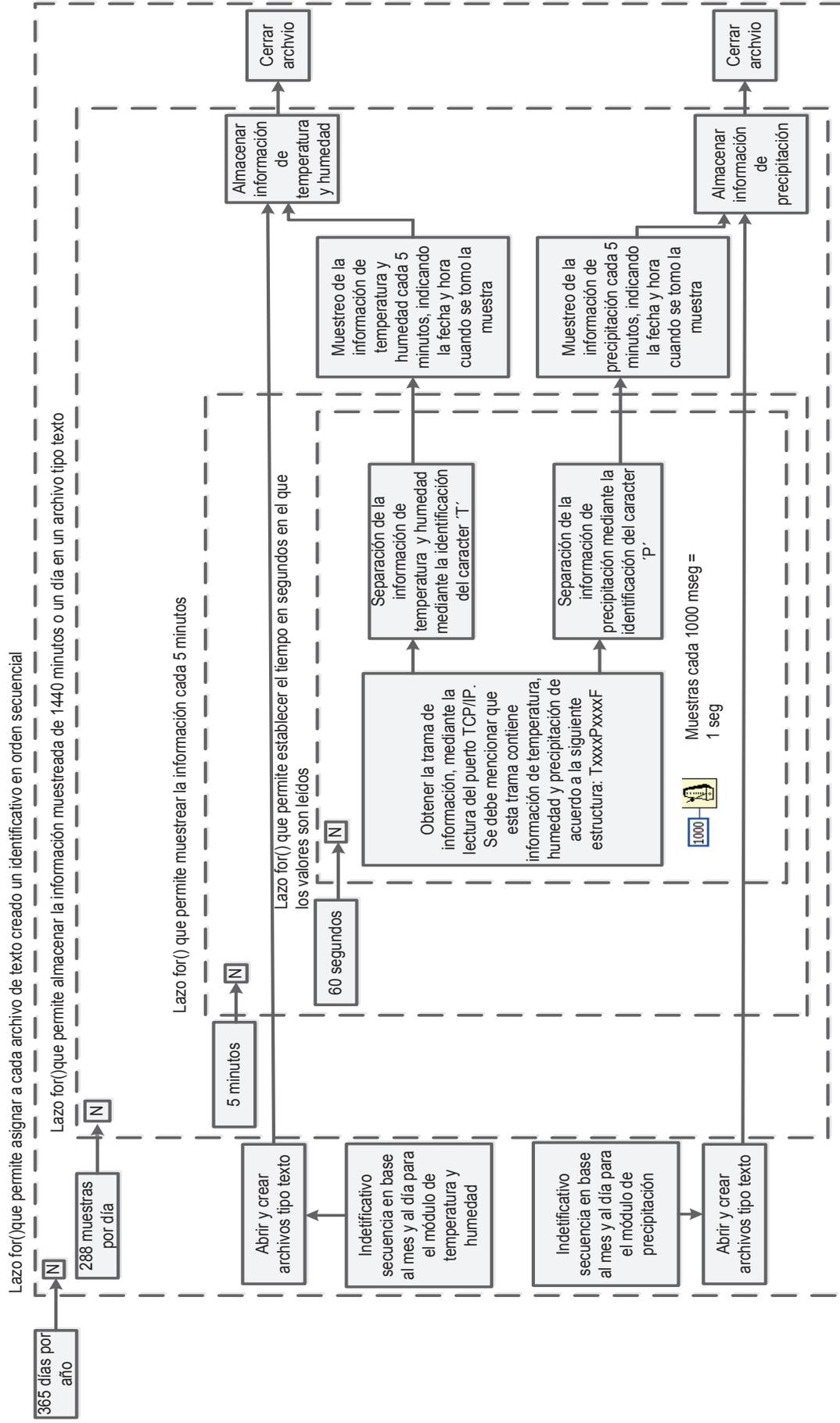
### **3.5.6. DESCRIPCIÓN Y DISEÑO DEL MÓDULO DE PRESENTACIÓN DE DATOS GPRS**

El módulo de presentación de datos GPRS se representa en el diagrama de bloques de la Figura 3.86<sup>34</sup>, en el cual se puede observar que el mismo está conformado por dos sistemas que permiten organizar la información por fecha y hora, seleccionar y presentar los datos de temperatura, humedad o precipitación de acuerdo a los requerimientos del usuario. Se debe mencionar que los sistemas de selección de información meteorológica, y presentación de datos individuales, son los mismos que se desarrollaron y se diseñaron en el módulo de presentación de datos meteorológicos del adquisidor de datos del módulo remoto. Se debe tomar en cuenta entonces, que los parámetros de temperatura, humedad y precipitación que se quieren organizar son muestreados al mismo instante cada 5 minutos.

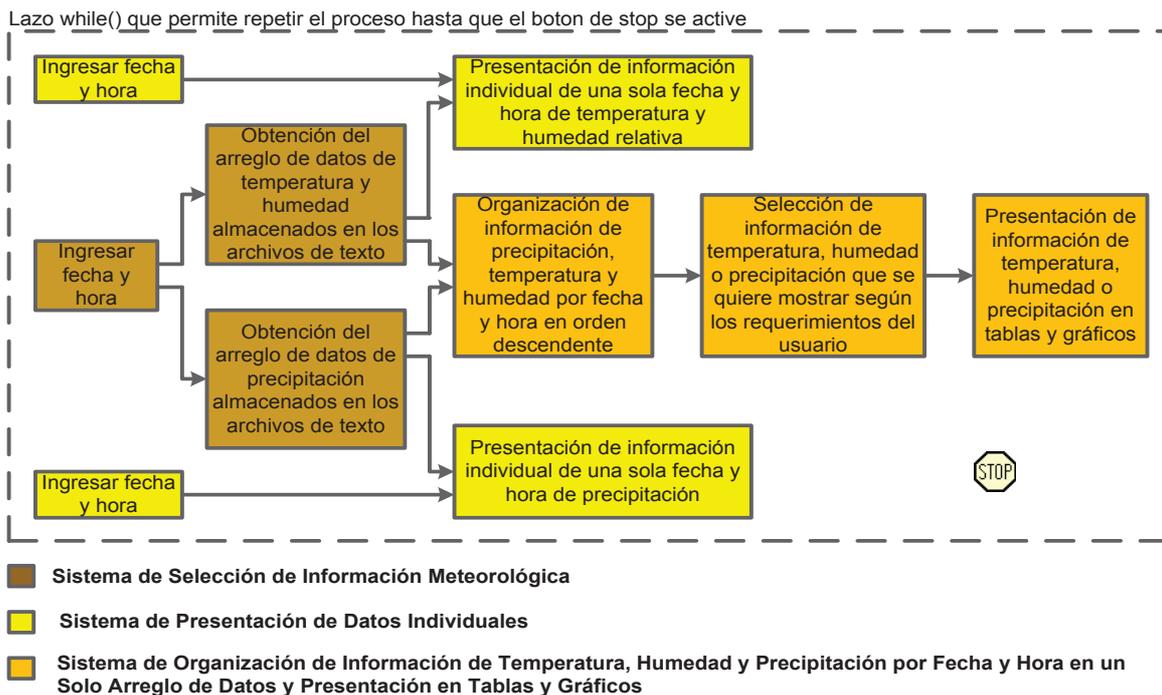
---

<sup>33</sup> El Anexo C7 presenta el diagrama de bloques detallado de la recopilación de datos GPRS.

<sup>34</sup> El Anexo C8 presenta el diagrama de bloques detallado de la presentación de datos GPRS.



**Figura 3.85 Diagrama de Flujo del Diseño del Módulo de Recopilación de Datos GPRS**  
Fuente: *Elaboración propia*



**Figura 3.86 Diagrama de Bloques del Módulo de Presentación de Datos GPRS**

*Fuente: Elaboración propia*

Por esta razón el arreglo de datos de estos tres parámetros organizados por fecha y hora en orden descendente, implica un proceso más sencillo de reubicación del parámetro de precipitación dentro de cada fila del arreglo de temperatura y humedad correspondiente a la misma fecha y hora. Este proceso se puede observar en el diagrama de bloques de la Figura 3.87<sup>35</sup>.

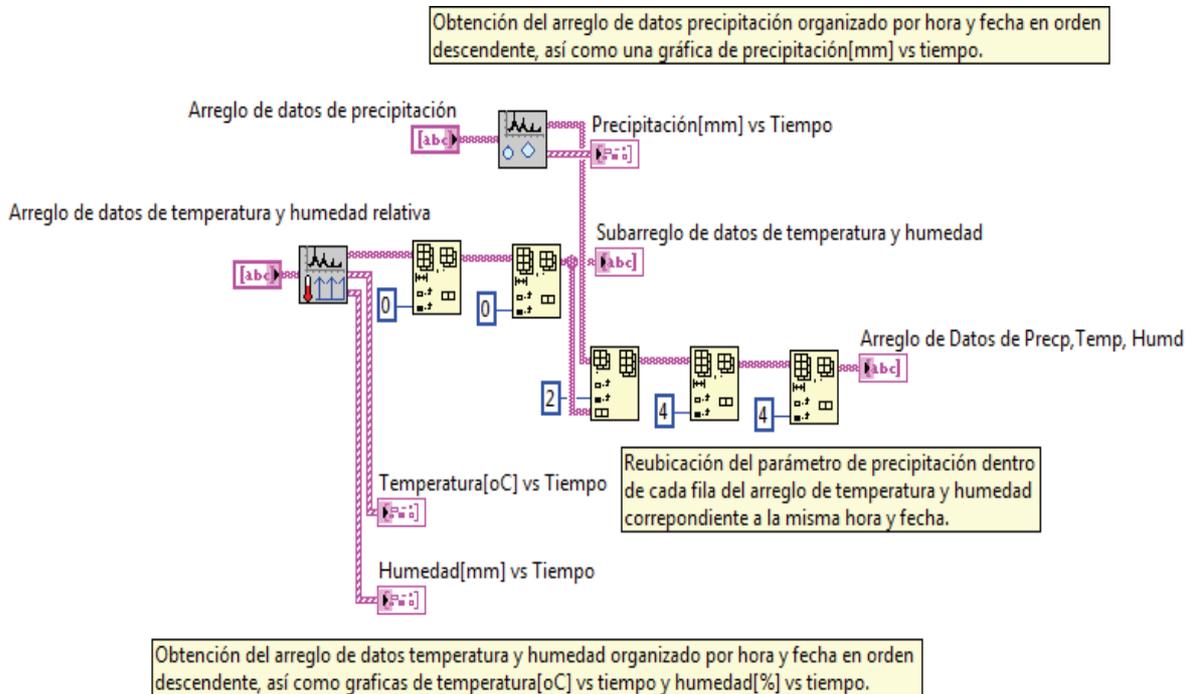
El resto de este sistema se mantiene tal como se diseñó originalmente para el módulo de adquisición de datos de la estación remota. En este sentido, se presenta en la Figura 3.88 la tabla de datos de la información obtenida.

### 3.5.7. VENTANA DE USUARIO

La ventana de usuario mostrada en la Figura 3.89 tiene como fin el de facilitar la gestión y manejo de la información GPRS adquirida, para lo cual, se debe integrar en un solo programa todos los módulos desarrollados y diseñados en secciones

<sup>35</sup> El Anexo C8.1 también se puede observar el diagrama de bloques para la obtención de gráficas y organización de la información GPRS.

anteriores, de tal forma que se ejecuten de forma simultánea. Las diferentes opciones que presenta esta ventana son:



**Figura 3.87** Proceso de Reubicación del Parámetro de Precipitación dentro del Arreglo de Temperatura y Humedad

Fuente: Elaboración propia utilizando del software Labview 2010

Fecha	Hora	Temperatura Relativa [oC]	Humedad Relativa [%]	Precipitación [mm]
15-octubre-2	14:53:51	18,0	56,0	0,0
15-octubre-2	14:58:52	18,0	57,4	0,0
15-octubre-2	15:03:52	17,7	58,3	0,0
15-octubre-2	15:08:52	17,5	59,6	0,0
15-octubre-2	15:13:52	17,4	61,0	0,0
15-octubre-2	15:18:52	17,3	60,7	0,0
15-octubre-2	15:23:52	17,5	58,8	0,0
15-octubre-2	15:28:52	18,0	55,9	0,0
15-octubre-2	15:33:52	17,4	57,9	0,0
15-octubre-2	15:38:52	17,0	59,7	0,0
15-octubre-2	15:43:52	17,0	59,8	0,0
15-octubre-2	15:48:52	17,0	58,9	0,0
15-octubre-2	15:53:52	16,7	61,6	0,0
15-octubre-2	15:58:52	16,6	62,5	0,0
15-octubre-2	16:03:52	16,3	66,2	0,0
15-octubre-2	16:08:52	15,9	67,7	0,0
15-octubre-2	16:13:52	15,8	66,4	0,0
15-octubre-2	16:18:52	15,8	66,6	0,0
15-octubre-2	16:23:52	15,8	66,8	0,0
15-octubre-2	16:28:52	15,9	65,5	0,0

**Figura 3.88** Arreglo de Datos GPRS Final de Temperatura, Humedad y Precipitación Organizado por Fecha y Hora en Orden Descendente

Fuente: Elaboración propia

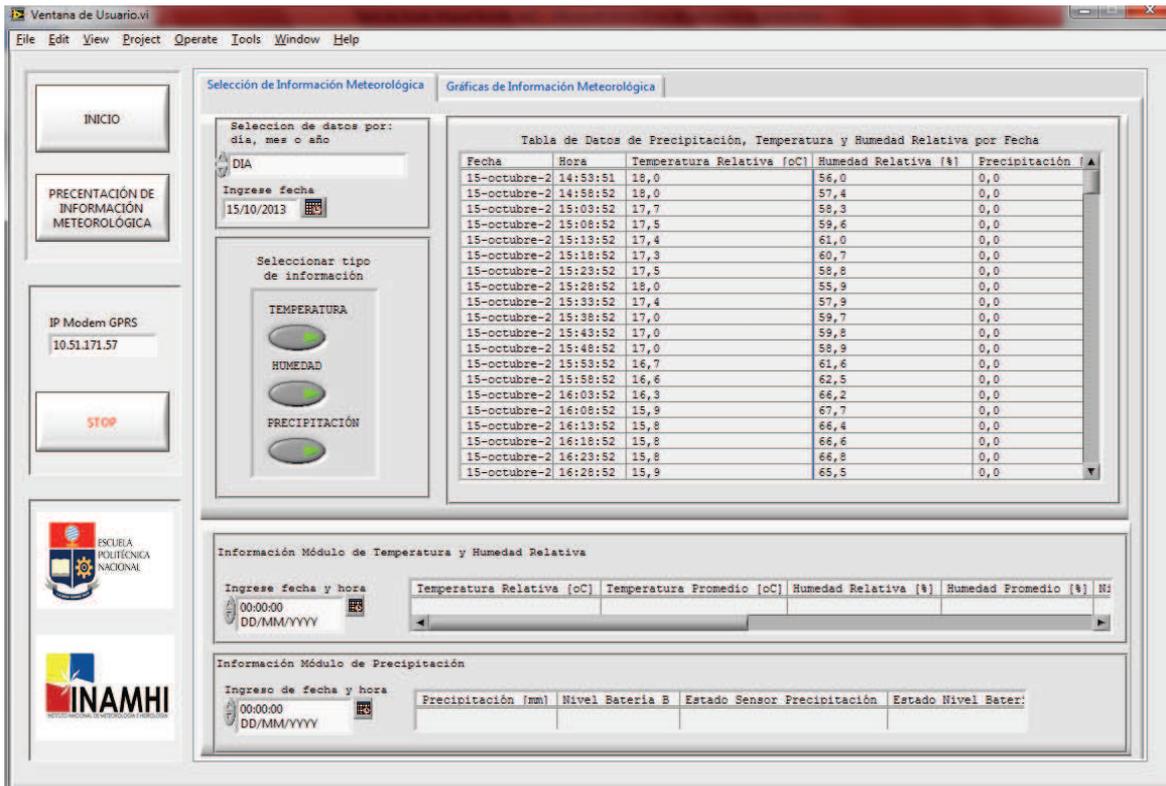
- a) Inicio.- Para presentación de la información general del módulo de adquisición de datos, tal como se muestra en la Figura 3.89.



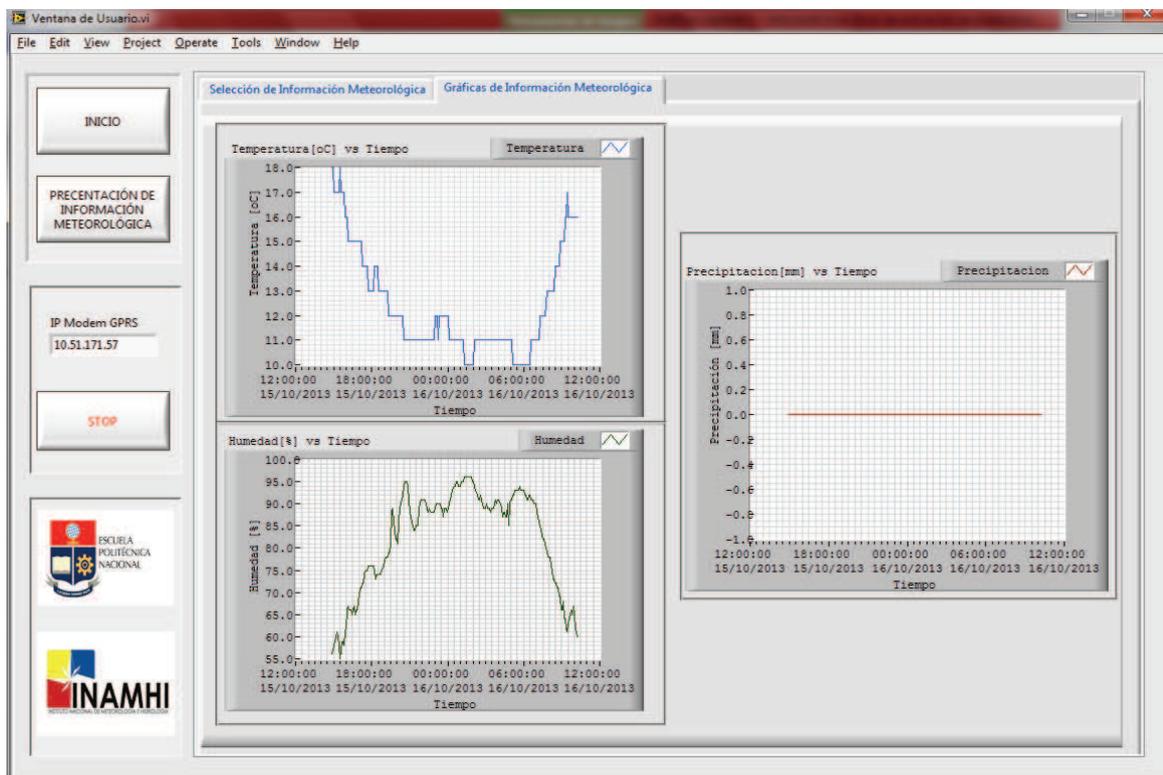
**Figura 3.89** Ventana de Inicio para Presentación de Información General del Módulo de Adquisición de Datos GPRS

*Fuente: Elaboración propia*

- b) Presentación de Información Meteorológica.- Para observación de la recopilación de información tanto de temperatura, humedad y precipitación, en diferentes pestañas tal como se muestra en la Figura 3.76.
- c) IP Modem GPRS.- Para ingresar la dirección IP del modem de una estación remota.
- d) STOP.- Para detener y cerrar todo el programa.



(a)



(b)

**Figura 3.90** Pestañas de la Ventana de Presentación de Datos GPRS donde se muestra: (a) Información de Temperatura, Humedad y Precipitación Ordenada por Fecha y Hora en Orden Descendente y (b) Gráficas de Temperatura, Humedad y Precipitación

Fuente: Elaboración propia

## **CAPÍTULO 4**

### **IMPLEMENTACIÓN Y PRUEBAS DEL PROTOTIPO**

#### **4.1. DISEÑO DE PLACAS**

El diseño de placas PCB (Printed Circuit Board), se realizó por medio del programa Altium Designer Summer 09, en el cual, se procedió inicialmente con el diagrama esquemático de los circuitos del módulo de temperatura y humedad relativa como del módulo de precipitación, para a continuación realizar el enrutamiento de los diferentes elementos de los circuitos, tomando en cuenta las siguientes características de la placas PCB:

- Tipo de placa: baquelita.
- Tipo de impresión: doble cara.
- Espesor de las placas: 2mm
- Ancho de las pistas de GND y 12V: 2mm para soportar aproximadamente 5A de intensidad de corriente.
- Ancho de las pistas: 1mm para soportar aproximadamente 2A de intensidad de corriente.

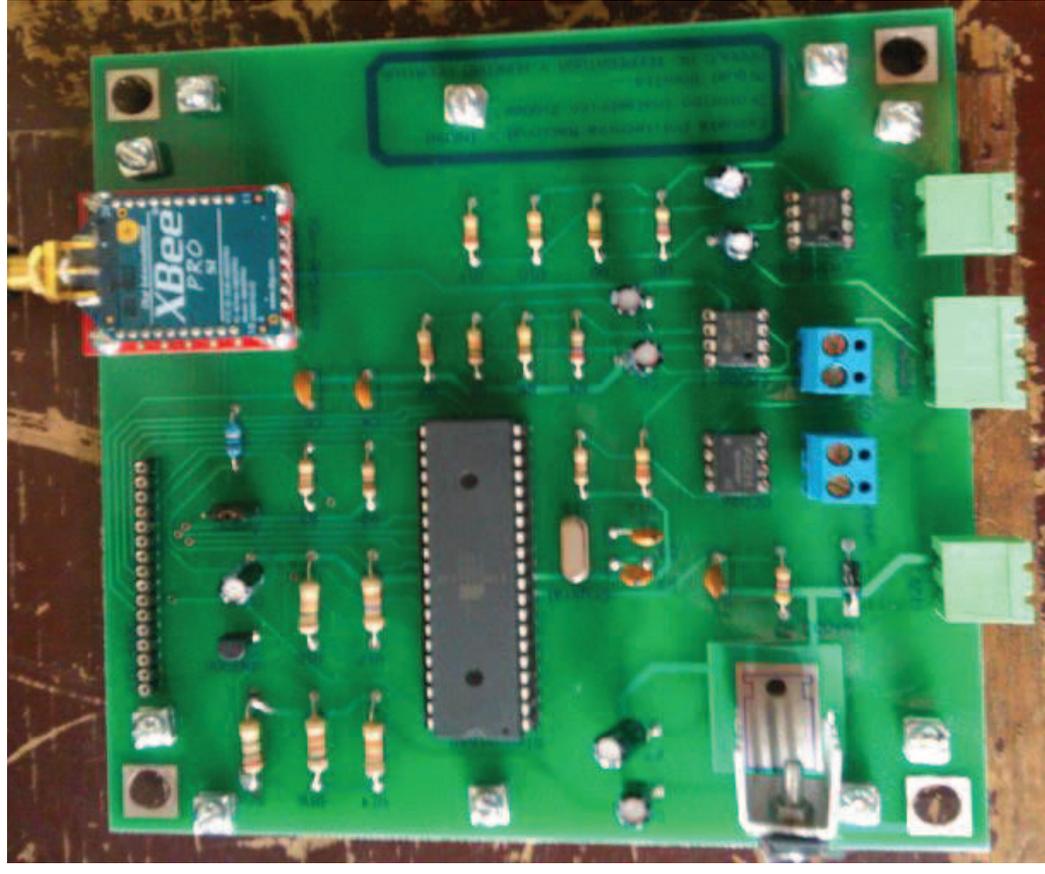
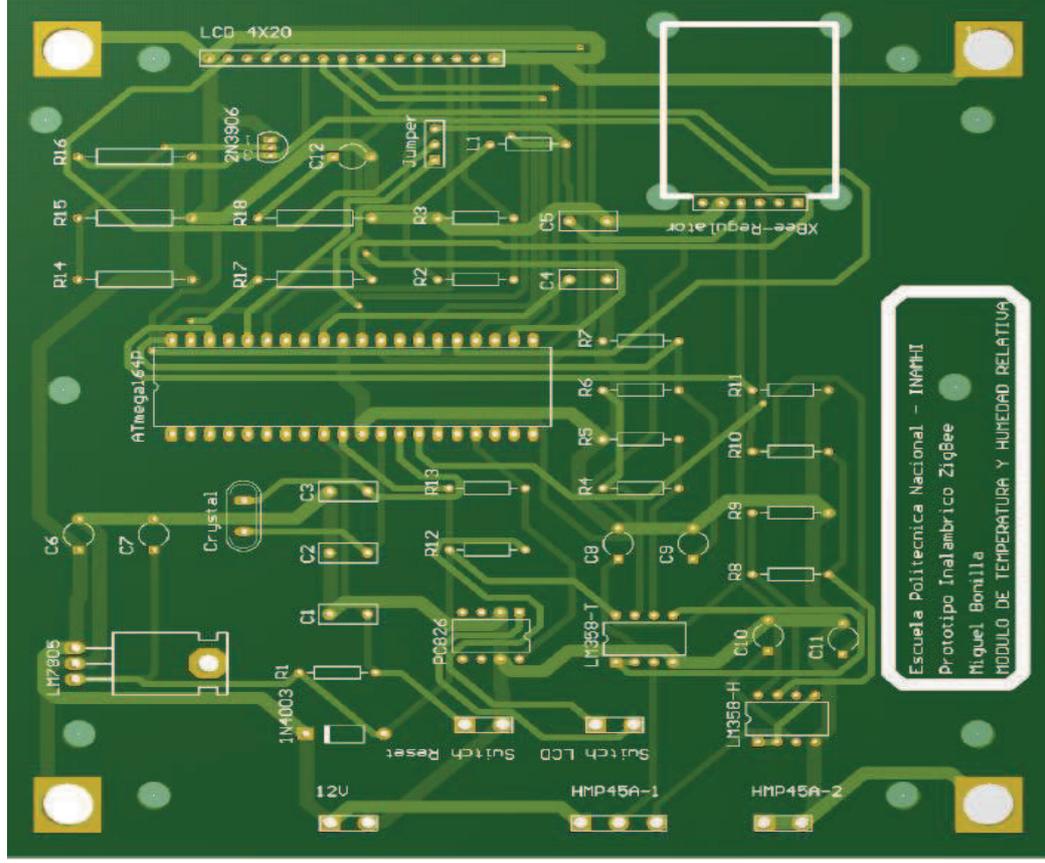
##### **4.1.1. DIAGRAMAS ESQUEMÁTICOS DE LAS PLACAS DISEÑADAS**

En esta sección se muestra en la Figura 4.1 la cara frontal del PCB y la placa con los correspondientes elementos soldados, del módulo de temperatura y humedad relativa.

De igual forma, se muestra en la Figura 4.2 la cara frontal del PCB y la placa con los correspondientes elementos soldados del módulo de precipitación.

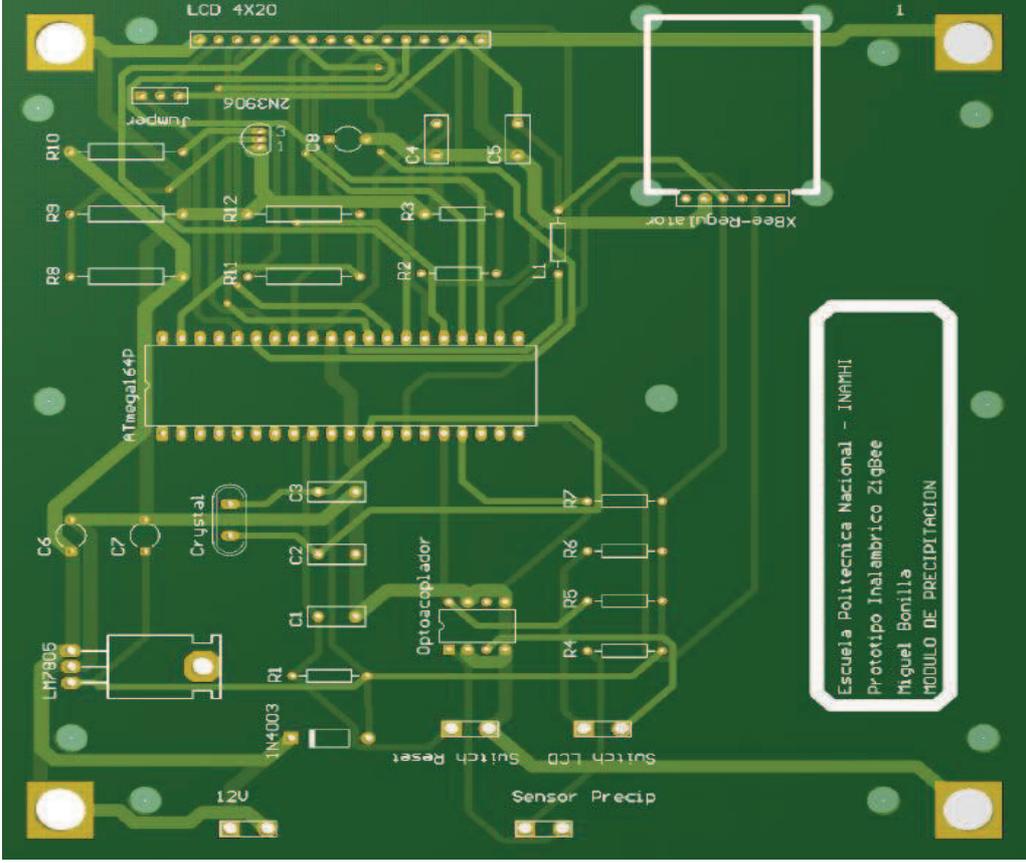
#### **4.2. CALIBRACIÓN DE LOS SENSORES**

La calibración debe ser realizada previamente a la instalación en campo y corresponde a los sensores de: temperatura, humedad y precipitación. Además, se corrigió errores de precisión presentados en el circuito interfaz de transductor.

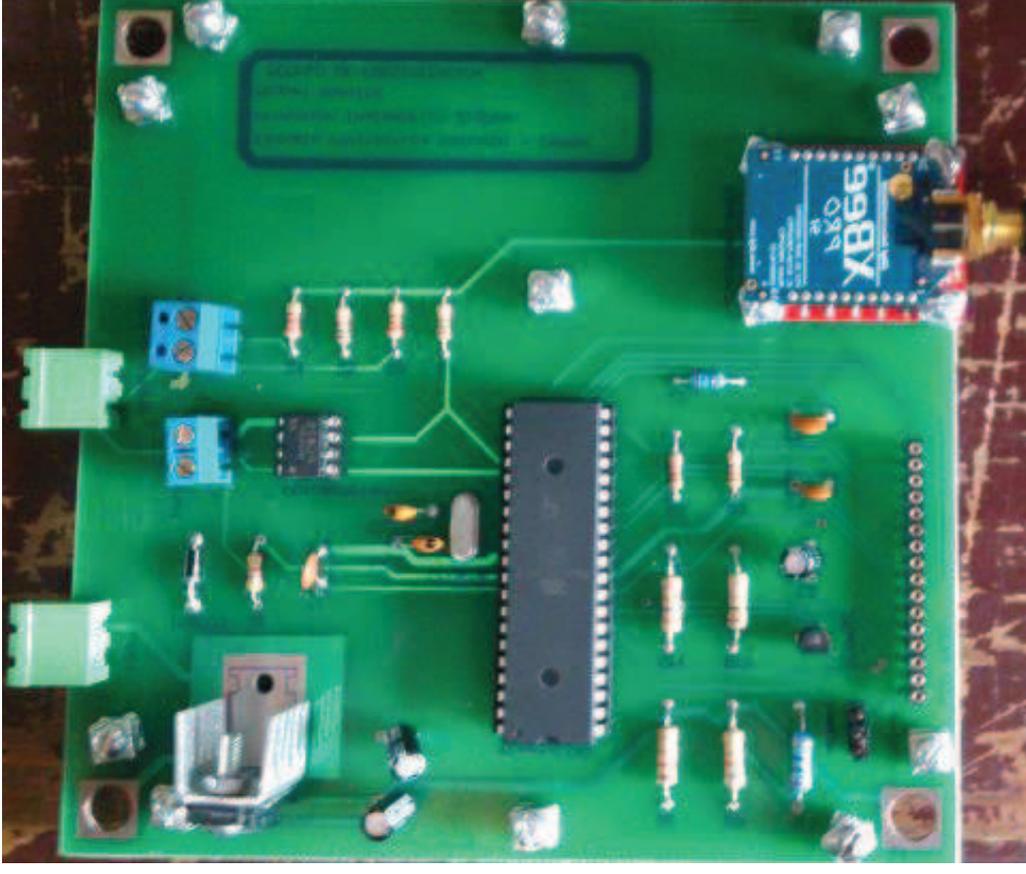


(a) Diagrama Esquemático de la Cara Frontal del PCB del Módulo de Temperatura y Humedad Relativa y (b) Placa con los Correspondientes Elementos soldados del Módulo de Temperatura y Humedad Relativa

Fuente: *Elaboración propia*



(a) Diagrama Esquemático de la Cara Frontal del PCB del Módulo de Precipitación y (b) Placa con los Correspondientes Elementos soldados del Módulo de Precipitación



(b)

Fuente: *Elaboración propia*

#### 4.2.1. CALIBRACIÓN DEL SENSOR DE TEMPERATURA HMP45A

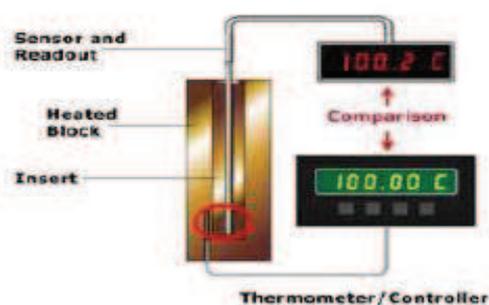
La calibración del sensor de temperatura HMP45A se realizó en el laboratorio de meteorología del INAMHI, por medio de la utilización de una cámara de pozo seco, tal como se muestra en la Figura 4.3, tomando en cuenta el procedimiento de calibración del manual de confirmación meteorológica en laboratorio.



**Figura 4.3** Calibración del Sensor de Temperatura HMP45A, en el Laboratorio de Metrología del INAMHI, por Medio de la Utilización de una Cámara de Pozo Seco

*Fuente: Elaboración propia*

En este sentido, la calibración de temperatura consiste en colocar un termómetro bajo prueba en conjunto con un termómetro patrón de referencia externo, dentro de un ambiente con temperatura estable y conocida, de tal manera que a continuación se procede a comparar entre la temperatura real del termómetro patrón externo y la temperatura de lectura que indica el termómetro bajo prueba tal como se muestra en la Figura 4.4. En caso de existir diferencias, que casi siempre las hay, éstas serán anotadas, con el propósito de realizar los correspondientes ajustes en el dispositivo de lectura, que en este caso le corresponde al módulo de adquisición de datos de temperatura y humedad relativa previamente diseñado [41].



**Figura 4.4 Proceso de Calibración de Temperatura**

Fuente: [41]

Además, el procedimiento de calibración incluye la toma de valores de voltaje del sensor de prueba en 7 puntos de medición tal como se muestra en la Tabla 4.1, de tal manera que se pueda realizar el correspondiente ajuste entre los valores de voltaje obtenidos y los valores de temperatura del termómetro patrón, consiguiendo como resultado los coeficientes de calibración de una ecuación lineal, adquirida mediante el método de mínimos cuadrados de regresión lineal tal como se muestra en la Figura 4.5, para lo cual se hace uso del software Statix que permite determinar los coeficientes de dichas ecuaciones.

**Tabla 4.1 Valores de Voltaje vs Temperatura del Sensor HMP45A**

Puntos de Medición	Valores de Voltaje de Temperatura del sensor HMP45A [mV]	Valores de Temperatura del Sensor HMP45A [°C]
1	305	-9.617
2	398	0.022
3	503	10.054
4	599	19.992
5	700	29.992
6	800	39.960
7	900	49.914

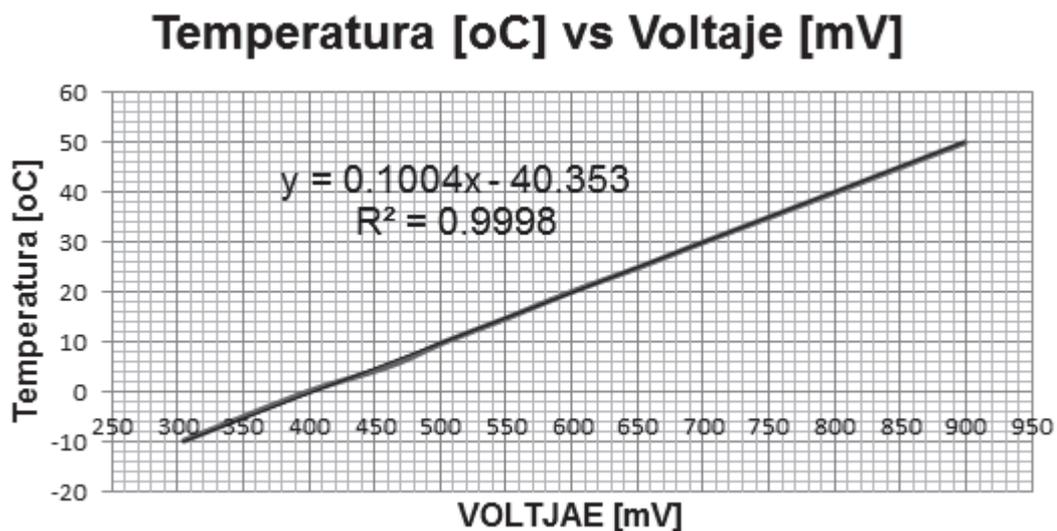
Fuente: Elaboración propia basada en información obtenida en el laboratorio del Meteorología del INAMHI

Se observa entonces que los coeficientes para medición de temperatura a ser ingresados en el software adquisidor de datos son:  $A = 0.1004$  y  $B = -40.535$

#### 4.2.2. CALIBRACIÓN DEL SENSOR DE HUMEDAD HMP45A

La calibración del sensor de humedad HMP45A se realizó en el laboratorio de meteorología del INAMHI, por medio de la utilización de una cámara de humedad

(tal como se muestra en la Figura 4.6), de manera similar al procedimiento de calibración del sensor de temperatura.



**Figura 4.5 Gráfica de Temperatura vs Voltaje y su correspondiente Ecuación Lineal de los Datos Obtenidos del Sensor de Temperatura HMP45A**

*Fuente: Elaboración propia basada en la utilización de Microsoft Excel*



**Figura 4.6 Calibración del Sensor de Humedad HMP45A, en el Laboratorio de Metrología del INAMHI, por Medio de la Utilización de una Cámara de Humedad**

*Fuente: Elaboración propia*

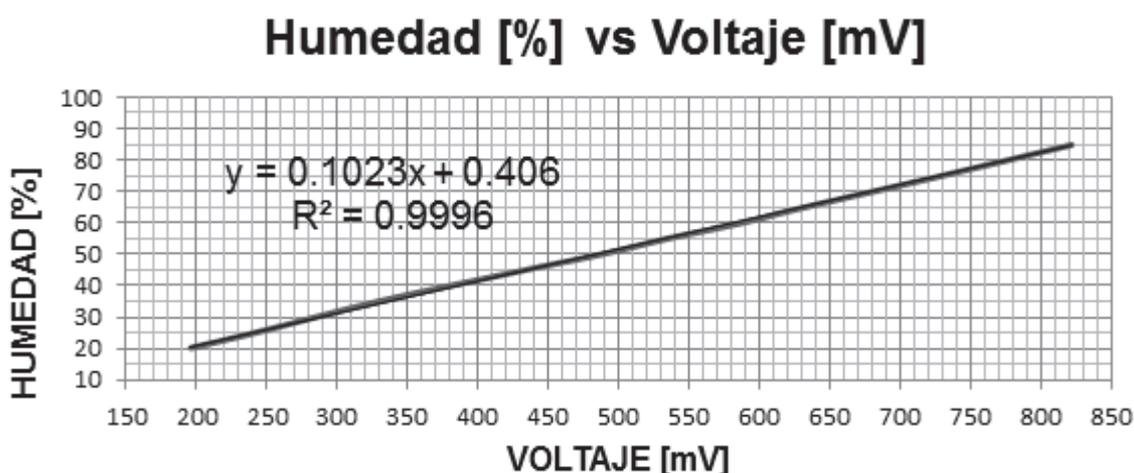
De igual forma, se incluye en el procedimiento de calibración la toma de valores de voltaje del sensor de prueba en 5 puntos de medición tal como se muestra en la Tabla 4.2, de tal manera que se pueda realizar el correspondiente ajuste entre

los valores de voltaje obtenidos y los valores de humedad del sensor patrón, consiguiendo como resultado los coeficientes de calibración de una ecuación lineal, adquirida mediante el método de mínimos cuadrados de regresión lineal tal como se muestra en la Figura 4.7, para lo cual se hace uso del software Statix que permite determinar los coeficientes de dichas ecuaciones.

**Tabla 4.2 Valores de Voltaje vs Humedad del Sensor HMP45A**

Puntos de Medición	Valores de Voltaje de Humedad del sensor HMP45A [mV]	Valores de Humedad del Sensor HMP45A [%]
1	196	20.16
2	487	49.92
3	587	59.95
4	822	84.85
5	730	75.02

*Fuente: Elaboración propia basada en información obtenida en el laboratorio del Meteorología del INAMHI*



**Figura 4.7 Gráfica de Humedad vs Voltaje y su Correspondiente Ecuación Lineal de los Datos Obtenidos del Sensor HMP45A**

*Fuente: Elaboración propia*

Se observa que los coeficientes para medición de humedad a ser ingresados en el software adquirente de datos son:  $a = 0.1023$  y  $b = 0.406$ .

#### 4.2.3. CALIBRACIÓN DEL SENSOR DE PRECIPITACIÓN TR-525M

El proceso de calibración [42] y ajuste del sensor de precipitación, se realizó en el laboratorio de meteorología del INAMHI, por medio del kit de calibración FC-525 de Texas Instrument, el cual contiene varios elementos:

- Vaso graduado con ranura de drenaje previamente perforada para una precisa medición del líquido.
- Embudo calibrador color negro anodizado.
- Múltiples boquillas de inyección para varias calibraciones. Incluye boquillas número 65 y número 70.
- Pines para la limpieza de cada boquilla de inyección.

#### 4.2.3.1. **Proceso de Calibración del Sensor de Precipitación TR-525M de Texas Instrument**

El proceso de calibración de acuerdo al manual de confirmación meteorológica para calibración de un sensor de precipitación de Texas Instrument [42], es el siguiente:

- 1) **Preparación del Sensor.-** Se debe inicialmente remover el colector de oro anodizado, y reemplazarlo con el embudo de calibración negro y la boquilla, que permitirán validar los resultados. Adicionalmente, se debe nivelar el sensor sobre una superficie lisa, que permita la recolección de datos de agua lluvia.
- 2) **Reseteo del Contador y del Registrador.-** Previo a la toma de medidas, se debe resetear el contador en cero y asegurarse de que el sensor, como el módulo de precipitación encargado de recolectar la información, se encuentren funcionando correctamente.
- 3) **Medición de Agua.-** Llenar un vaso graduado con 99ml de agua.
- 4) **Vertido de Agua en el Embudo de Calibración.-** Derramar toda el agua del vaso graduado, sobre el embudo de calibración con precisión y cuidado.
- 5) **Conteo de Basculaciones.-** Finalmente se debe realizar el conteo de basculaciones registradas por el módulo de precipitación. El número de basculaciones previstas para una cantidad de 99ml de agua es de 21, tomando en cuenta que en caso de registrar un mayor o menor número de basculaciones, el sensor de precipitación debe ser ajustado.

#### 4.2.3.2. Proceso de Ajuste del Sensor de Precipitación TR-525M

El proceso de ajuste de acuerdo al manual de confirmación meteorológica para calibración de un sensor de precipitación de Texas Instrument [42] , en caso de que la recopilación de basculaciones no sea el mismo que indica el proceso de calibración, es el siguiente:

- 1) **Ubicar los Tornillos de Ajuste.-** Los tornillos de ajuste del sensor de precipitación TR-525M, se encuentran en la sección inferior, para lo cual es necesario colocar el sensor bocabajo. Se debe notar que el ajuste debe ser realizado mediante el giro de ambos tornillos en igual cantidad, para mantener la caja de inflexión balanceada.
- 2) **Ajuste de los Tornillos de Calibración.-** En el caso de que se requiera incrementar la lectura del indicador de lluvia, se debe girar los tornillos en sentido horario. Caso contrario, para disminución de la lectura, se debe girar los tornillos en sentido anti horario. Como regla general,  $\frac{1}{4}$  de giro de ambos tornillos representa aproximadamente 1 cuenta.
- 3) **Recalibrar el Sensor.-** Finalmente, se realiza el procedimiento de calibración previamente señalado hasta obtener los valores de basculaciones requeridos.

### 4.3. PRUEBAS Y RESULTADOS

En esta sección se realizan diferentes tipos de pruebas que garanticen el correcto funcionamiento del prototipo, como también la validez de la información meteorológica recopilada, procesada y adquirida.

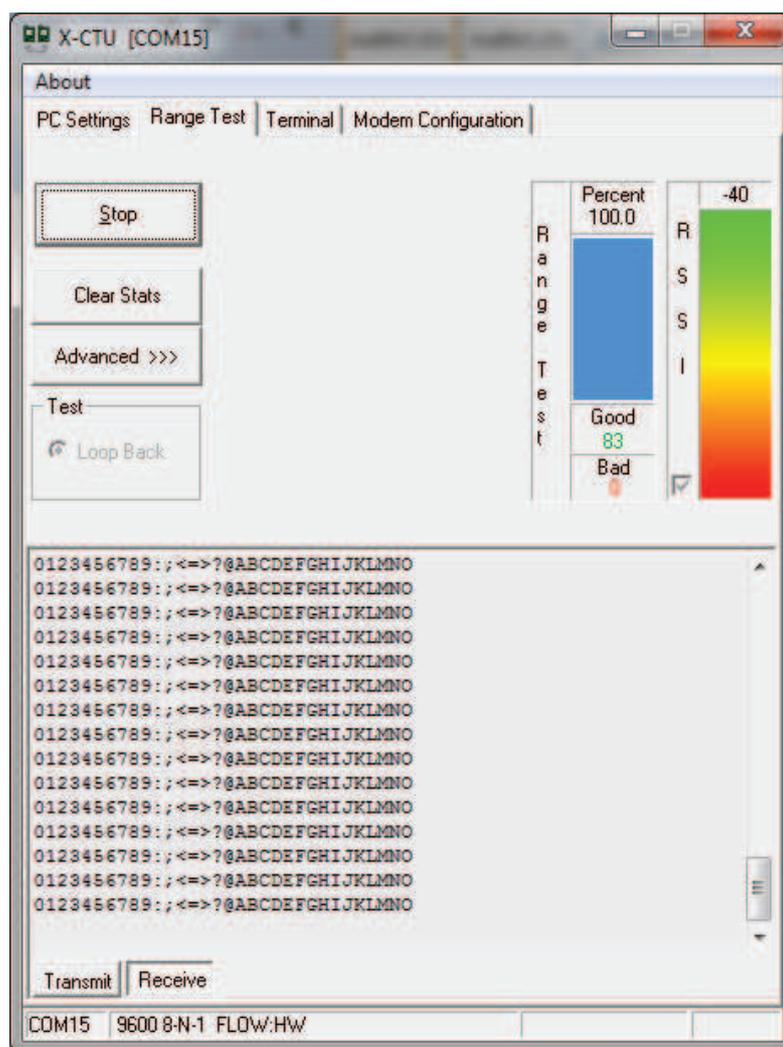
#### 4.3.1. PRUEBAS DE RECEPCIÓN RSSI

Las pruebas realizadas corresponden al nivel de recepción RSSI entre un módulo coordinador y un dispositivo final en una topología punto a punto en línea de vista, mediante la utilización de antenas tipo goma omnidireccional de media longitud de onda y 2.4GHz de frecuencia. De esta forma, se presenta en la Tabla 4.3 los resultados obtenidos en el software X-CTU, mediante la utilización de la ventana de verificación de nivel RSSI mostrada en la Figura 4.8.

**Tabla 4.3 Resultados de Nivel de Señal RSSI en -dBm en una Topología Punto a Punto entre dos Dispositivos XBee-PRO**

Distancia de Prueba	Nivel de Señal de Recepción RSSI	Cantidad de Paquetes Recibidos	Cantidad de Paquetes Rechazados	Porcentaje de Recepción de paquetes
10m	-40dBm	100	0	100%
20m	-40dBm	200	0	100%
30m	-40dBm	300	0	100%
40m	-40dBm	400	0	100%
50m	-41dBm	500	0	100%
60m	-42dBm	600	10	100%
70m	-48dBm	700	10	100%
80m	-41dBm	800	10	98.9%
90m	-55dBm	900	10	98.9%
96.5m	-54dBm	10000	10	98.9%

*Fuente: Elaboración propia*

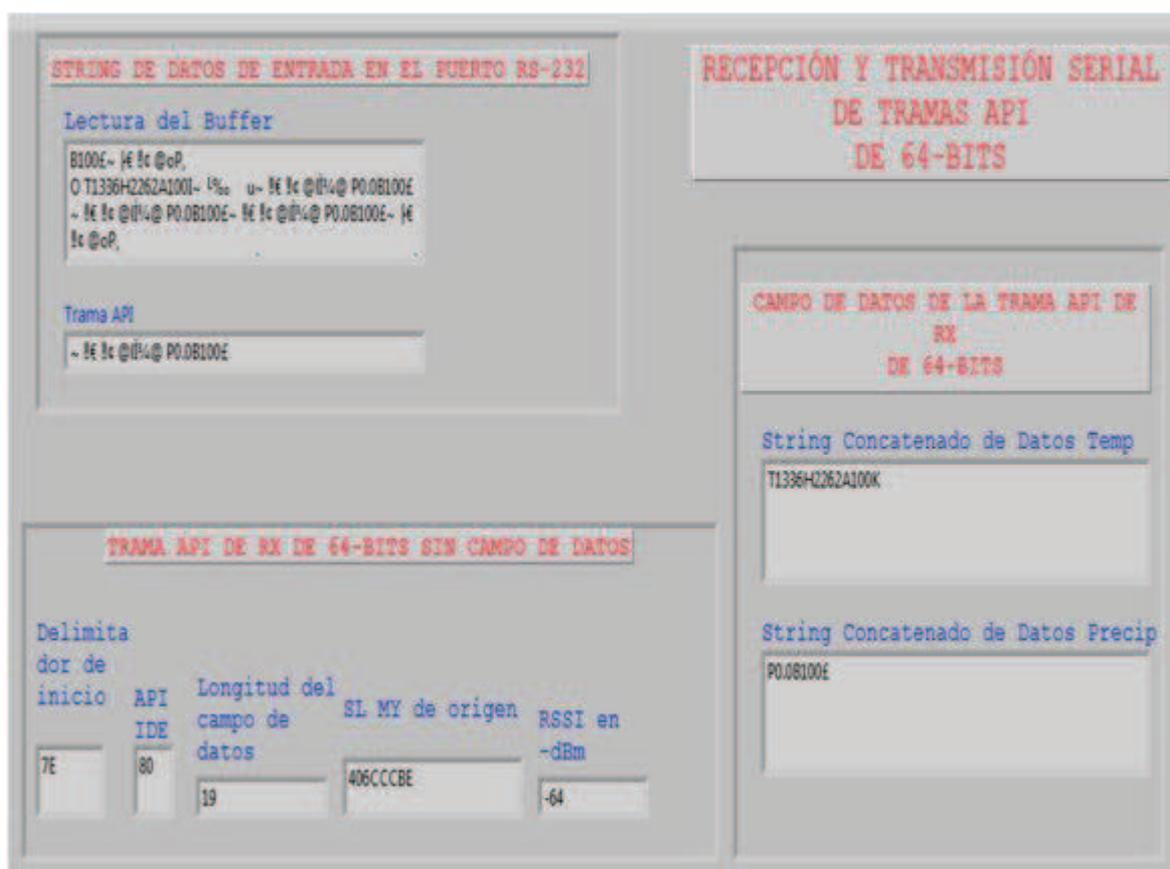


**Figura 4.8 Ventana de Verificación de Nivel de Señal RSSI del Software X-CTU**

*Fuente: Elaboración propia*

### 4.3.2. PRUEBAS DE FUNCIONAMIENTO DEL PROTOTIPO

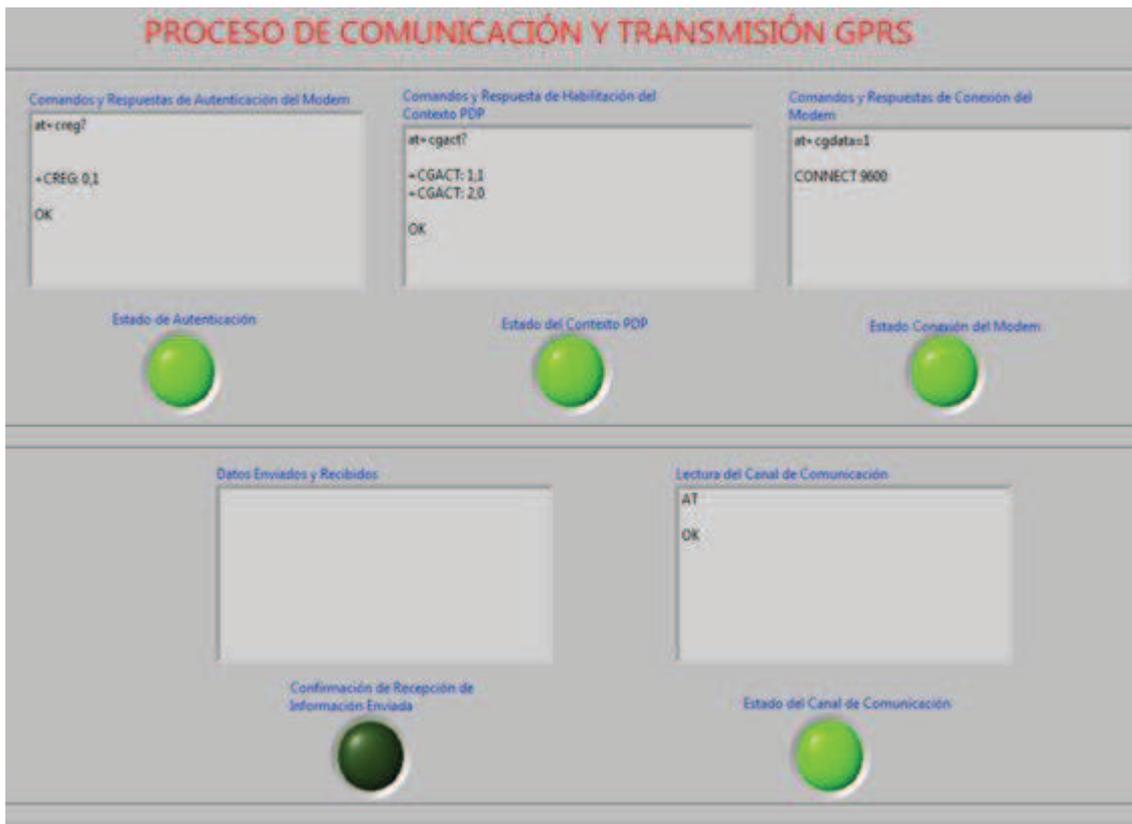
El funcionamiento del prototipo se verifica mediante la observación de la correcta ejecución de los instrumentos virtuales correspondientes a la transmisión y recepción serial API de la información del módulo de temperatura y humedad relativa como del módulo de precipitación, y que a su vez se muestra en la Figura 4.9.



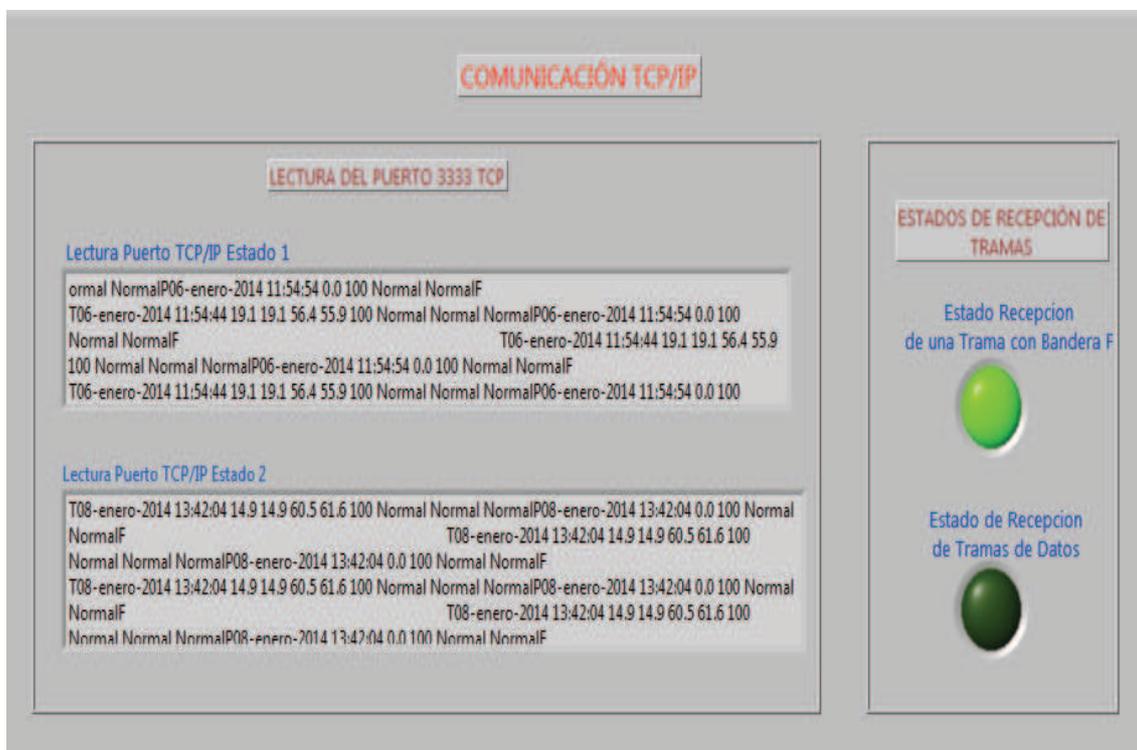
**Figura 4.9 Instrumento Virtual de Transmisión y Recepción de Tramas API de Información de Temperatura, Humedad y Precipitación**

*Fuente: Captura de pantalla del instrumento virtual de Tx/Rx API*

En cuanto al procesamiento de la información de temperatura, humedad y precipitación enviada vía GPRS, se verifica mediante la observación del correcto funcionamiento de los instrumentos virtuales de transmisión, recepción y presentación de datos GPRS del módulo de adquisición de datos de la estación remota y que se muestra en la Figura 4.10, así como del correspondiente instrumento virtual del servidor de información meteorológica localizado en las oficinas del INAMHI Quito, y que se muestra en la Figura 4.11.



**Figura 4.10 Transmisión de Información Meteorológica vía GPRS del Módulo de Adquisición de Datos**  
 Fuente: Captura de pantalla del instrumento virtual de comunicación y transmisión GPRS



**Figura 4.11 Recepción de Información Meteorológica vía GPRS en el Servidor localizado en el INAMHI Quito**

Fuente: Captura de pantalla del instrumento de comunicación TCP/IP del Servidor de Datos

### 4.3.3. PRUEBAS DE VALIDACIÓN DE INFORMACIÓN DE TEMPERATURA, HUMEDAD Y PRECIPITACIÓN

La validación de la información de temperatura y humedad, se realizó mediante un proceso de comparación de los valores obtenidos por el módulo de adquisición de datos, con los valores registrados por la estación Logotronic Ñaquito. Cabe destacar, que ambas estaciones se encuentran ubicadas en las instalaciones del INAMHI Quito y que los sensores de ambas estaciones se encuentran calibrados, luego de haber ingresado los coeficientes de calibración.

En este sentido, se procedió a la toma de valores de temperatura y humedad durante 2 días correspondientes a el 4 y 5 de enero del 2014, obteniendo una observación de 2880 valores de temperatura, humedad y precipitación, lo cual significa el registro de una muestra cada minuto durante 48 horas.

#### 4.3.3.1. Resultados de la Comparación de la Información Obtenida de Temperatura

Una vez adquirida la muestra de 2880 valores, se procedió a la obtención del error absoluto de cada valor de acuerdo a la Ecuación 4.1, en la cual  $x$  representa los valores obtenidos del módulo de temperatura y humedad relativa y  $x_0$  representa los valores de referencia de la estación Logotronic del INAMHI. De igual manera, se procedió a la consolidación de la Tabla 4.4 de distribución de frecuencia de la información de temperatura.

$$\Delta x = |x - x_0|$$

Ecuación 4.1

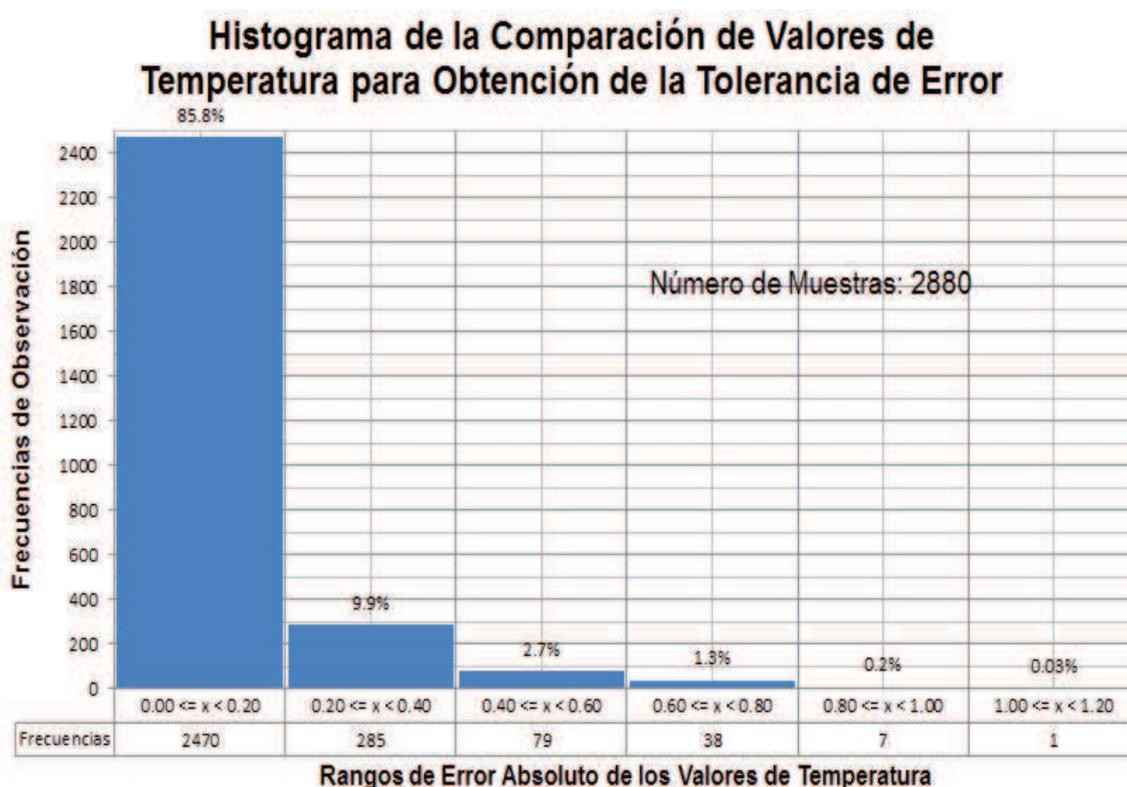
En función de los datos de la Tabla 4.4, se muestra los resultados en la Figura 4.12, en la cual se puede observar que 85.8 % de las muestras presentan un rango de diferencia de: 0°C - 0.2°C con respecto a los valores de referencia, lo cual indica que la medición de información dentro de este rango es óptima. El 9.9% de las muestras se encuentran en un rango de diferencia de: 0.2°C - 0.4°C, cuyo valor superior representa la tolerancia de error que el INAMHI garantiza al calibrar un sensor de temperatura. En este sentido se observa que 95.4% de

todos los valores muestreados se encuentran dentro del rango de tolerancia de error de:  $x \pm 0.2^{\circ}\text{C}$ , que el INAMHI debe garantizar en la medición de temperatura.

**Tabla 4.4 Distribución de Frecuencia para los Datos de Error Absoluto de la Comparación de Valores de Temperatura**

Intervalos de Clase de Rangos del Error Absoluto	Frecuencia	Frecuencia Relativa	Frecuencia Relativa Acumulada	Porcentaje con Respecto al Total de Muestras
$0.00 \leq x < 0.20$	2470	0.8576	0.8576	85.8%
$0.20 \leq x < 0.40$	285	0.0990	0.9566	9.9%
$0.40 \leq x < 0.60$	79	0.0274	0.9840	2.7%
$0.60 \leq x < 0.80$	38	0.0132	0.9972	1.3%
$0.80 \leq x < 1.00$	7	0.0024	0.9997	0.2%
$1.00 \leq x < 1.20$	1	0.0003	1.0000	0.03%
<b>Número de Muestras:</b>	2880		<b>Total Porcentaje:</b>	100%

Fuente: Elaboración propia en base a la observación de 2880 valores obtenidos



**Figura 4.12 Histograma de Comparación de Valores de Temperatura para Obtención del Error Absoluto de la Muestra de Valores Observados**

Fuente: Elaboración propia en base a la recolección de datos de dos días consecutivos

Por otro lado, 4.6% de todos los valores muestreados de temperatura se encuentran sobre el umbral de tolerancia de error. Por lo tanto, estos valores deben ser descartados, tomando en cuenta que la mayor parte de ellos se encuentran en un rango de:  $0.4^{\circ}\text{C} - 0.6^{\circ}\text{C}$ , correspondiente al 2.7%.

#### 4.3.3.2. Resultados de la Comparación de Información Obtenida de Humedad

Al igual que los resultados obtenidos de temperatura, la comparación de los valores de humedad se realizan de forma similar. En este sentido, los datos de la Tabla 4.5 resumen la recopilación de las 2880 muestras registradas durante 2 días consecutivos. Por otro lado, la Figura 4.13 muestra los resultados adquiridos para este caso. Se puede observar entonces, que el 78.6% de las muestras presentan un rango de diferencia entre: 0% - 1% con respecto a los valores de referencia, lo cual indica que solo este porcentaje de valores entra dentro del rango de medición óptimo. El 13.5% de las muestras se encuentran en un rango de diferencia de: 1% - 1.4%, cuyo valor superior representa la tolerancia de error máxima que el INAMHI garantiza al calibrar sensores de humedad. Por tanto, se observa que el 92.1% del total de muestras recopiladas se encuentran dentro del rango de tolerancia de  $x \pm 1.4\%$ , que el INAMHI debe garantizar.

**Tabla 4.5 Distribución de Frecuencia para los Datos de Error Absoluto de la Comparación de Valores de Humedad**

Intervalos de Clase de Rangos del Error Absoluto	Frecuencia	Frecuencia Relativa	Frecuencia Relativa Acumulada	Porcentaje con Respecto al Total de Muestras
0.00 $\leq$ x < 1.00	2265.0	0.7865	0.7865	78.6%
1.00 $\leq$ x < 1.40	390.0	0.1354	0.9219	13.5%
1.40 $\leq$ x < 2.00	140.0	0.0486	0.9705	4.9%
2.00 $\leq$ x < 3.00	65.0	0.0226	0.9931	2.3%
3.00 $\leq$ x < 4.00	16.0	0.0056	0.9986	0.6%
4.00 $\leq$ x < 5.50	4.0	0.0014	1.0000	0.1%
<b>Número de Muestras:</b>	2880		<b>Total Porcentaje:</b>	100%

*Fuente: Elaboración propia en base a la observación de 2880 valores obtenidos*

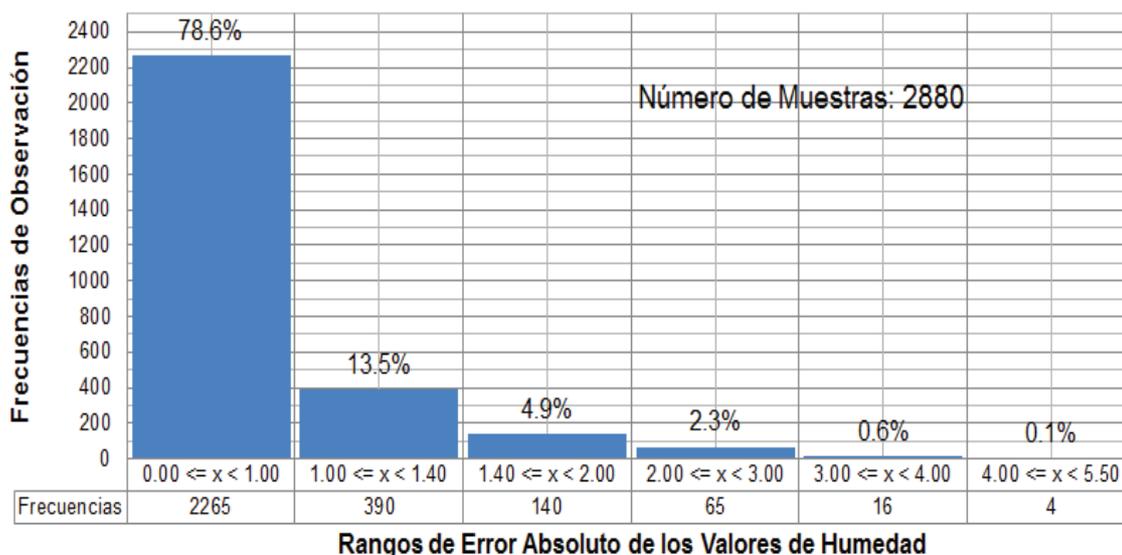
Por otro lado, 7.9% de todos los valores muestreados de humedad se encuentran sobre el umbral de tolerancia de error. Por lo tanto, estos valores deben ser descartados, tomando en cuenta que la mayor parte de ellos se encuentran en un rango de: 1.4% - 2%, correspondiente al 4.9%.

#### 4.3.3.3. Resultados de la Comparación de Información Obtenida de Precipitación

En cuanto al sensor de precipitación, se cumplió con el rango de error permisible por medio del conteo de 21 basculaciones, luego de haber vertido 99 ml de agua en el embudo de calibración del sensor de precipitación previamente nivelado.

De igual manera, se obtuvieron las 2880 muestras. Sin embargo, no se registraron basculaciones en la estación de referencia de Iñaquito, ni en el módulo de precipitación.

### Histograma de la Comparación de Valores de Humedad para Obtención de la Tolerancia de Error



**Figura 4.13** Histograma de Comparación de Valores de Humedad para Obtención del Error Absoluto de la Muestra de Valores Observados

*Fuente: Elaboración propia en base a la recolección de datos de dos días consecutivos*

#### 4.4. INSTALACIÓN DEL PROTOTIPO EN CAMPO

La infraestructura de soporte necesaria para instalar el prototipo, está conformada por todas las estructuras necesarias para protección del Datalogger, colocación y sustento de sensores, protecciones de cables y sensores, y montaje de equipos. De esta forma, se enumera los siguientes ítems:

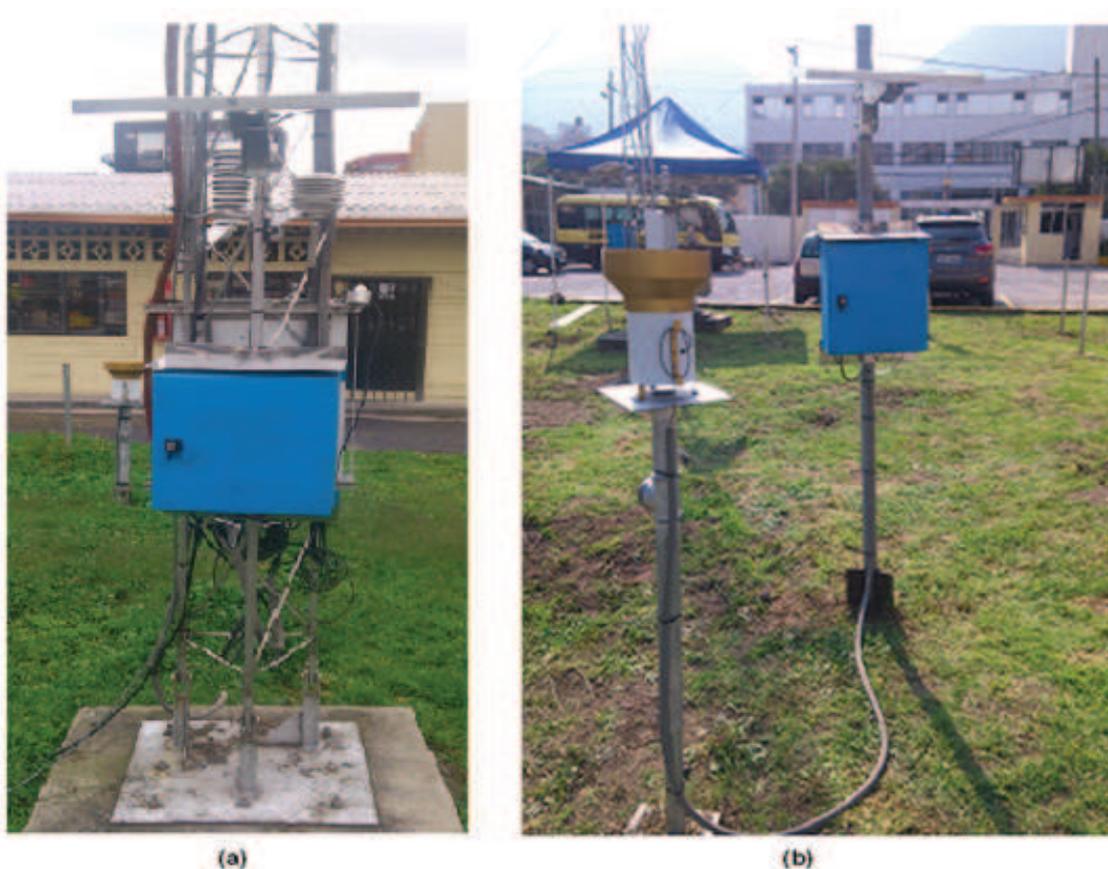
- Caja nema mostrada en la Figura 4.14, para protección de equipos eléctricos y electrónicos ubicados en exteriores.
- Pantalla de protección del sensor de humedad y temperatura relativa, para evitar errores de radiación solar o impurezas del ambiente e inclusive por filtración de agua.
- Postes y torres para soporte de sensores, caja nema y pluviómetros.
- Elementos de sujeción y protecciones de cables y de las conexiones al gabinete o caja nema.



**Figura 4.14** Caja de Protección del Módulo de Temperatura y Humedad Relativa como del Módulo de Precipitación

*Fuente: Fotografía tomada en las oficinas del INAMHI Quito*

Por tanto, se presenta en la Figura 4.15 la instalación en campo tanto del módulo de temperatura y humedad relativa, como el módulo de precipitación.



**Figura 4.15** Módulos Instalados en campo de: (a) Temperatura y Humedad Relativa, (b) Precipitación

*Fuente: Fotografías tomadas en las instalaciones del INAMHI Quito*

## 4.5. COSTOS REFERENCIALES DEL PROTOTIPO

**Tabla 4.6** Especificaciones de Costos de Elementos y Dispositivos del Módulo de Temperatura y Humedad Relativa

Descripción	Cantidad	Valor Unitario \$	Valor Total \$
<b>Circuitos Integrados</b>			
Microcontrolador ATmega164P con encapsulado tipo PDIP	1	7.80	7.8
Módulo de transmisión XBee-PRO 802.15.4 OEM RF	1	86.41	86.41
XBee-Regulator	1	20.00	20
Display LCD 4x20 con backlight	1	21.42	21.42
Regulador de voltaje 5V LM7805	1	0.38	0.38
Optoacoplador PC826	1	0.40	0.40
<b>Antena y Elementos de Energización</b>			
Batería 12V	1	73.33	73.33
Antena Rubber Ducky 2.4 GHz, 6" half wave dipole, omni directional, articulating RPSMA male (2.1 dBi)	1	68.50	68.50
<b>Elementos Activos y Pasivos</b>			
Diodo 1N4003	1	0.07	0.07
Transistor 2N3906	2	0.08	0.16
Resistencia 1/2W	8	0.03	0.24
Resistencia 1/4W	6	0.02	0.12
Capacitor cerámico	7	0.07	0.49
Inductor	1	0.53	0.53
Crystal	1	0.54	0.54
<b>Conectores y Cables</b>			
Bornera 2 pines	2	0.22	0.44
Terminal macho hembra 2 pines	2	1.45	2.9
Terminal macho hembra 3 pines	1	2.19	2.19
Zócalo 40 pines	1	0.26	0.26
Zócalo 8 pines	1	0.07	0.07
Espadín maquinado hembra	32	0.02	0.64
Cables de conexión 15cm	16	0.48	7.68
<b>Elementos de Implementación y Ensamblaje</b>			
Placa PCB	1	38.80	38.80
Caja acrílica	1	30	30
Pulsador switch 2 terminales	2	0.43	0.86
<b>Total Costo Módulo de Temperatura y Humedad Relativa</b>			<b>\$364.23</b>

*Fuente: Elaboración propia*

**Tabla 4.7 Especificaciones de Costos de Elementos y Dispositivos del Módulo de Precipitación**

<b>Descripción</b>	<b>Cantidad</b>	<b>Valor Unitario \$</b>	<b>Valor Total \$</b>
<b>Circuitos Integrados</b>			
Microcontrolador ATmega164P con encapsulado tipo PDIP	1	7.80	7.8
Módulo de transmisión XBee-PRO 802.15.4 OEM RF	1	86.41	86.41
XBee-Regulator	1	20.00	20
Display LCD 4x20 con backlight	1	21.42	21.42
Regulador de voltaje 5V LM7805	1	0.38	0.38
Optoacoplador PC826	2	0.40	0.40
<b>Antena y Elementos de Energización</b>			
Batería 12V	1	73.33	73.33
Antena Rubber Ducky 2.4 GHz, 6" half wave dipole, omni directional, articulating RPSMA male (2.1 dBi)	1	68.50	68.50
<b>Elementos Activos y Pasivos</b>			
Diodo 1N4003	1	0.07	0.07
Transistor 2N3906	2	0.08	0.16
Resistencia 1/2W	8	0.03	0.24
Resistencia 1/4W	8	0.02	0.16
Capacitor cerámico	7	0.07	0.49
Inductor	1	0.53	0.53
Crystal	1	0.54	0.54
<b>Conectores y Cables</b>			
Bornera 2 pines	2	0.22	0.44
Terminal macho hembra 2 pines	2	1.45	2.90
Zócalo 40 pines	1	0.26	0.26
Zócalo 8 pines	1	0.07	0.07
Espadín maquinado hembra	16	0.02	0.64
Cables de conexión 15cm	16	0.48	7.68
<b>Elementos de Implementación y Ensamblaje</b>			
Placa PCB	1	38.80	38.80
Caja acrílica	1	30	30
Pulsador switch 2 terminales	2	0.43	0.86
<b>Total Costo Módulo de Precipitación</b>			<b>\$362.48</b>

*Fuente: Elaboración propia basado en facturas y proformas*

Los costos referenciales que se describen en esta sección, se clasifican de acuerdo al tipo de módulo del prototipo, y en consecuencia a los dispositivos y elementos que cada uno contenga.

#### 4.5.1. COSTO REFERENCIAL DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA

Los costos referenciales de este módulo observados en la Tabla 4.6, se basan en los elementos y dispositivos que entran dentro del diseño y que no han sido provistos por INAMHI. En este contexto, se debe indicar que tanto el sensor HMP45A, panel solar y controlador de carga no han sido tomados en cuenta, ya que fueron proporcionados por el INAMHI.

#### 4.5.2. COSTO REFERENCIAL DEL MÓDULO DE PRECIPITACIÓN

En este caso los costos se muestran en la Tabla 4.7. Se debe considerar, al igual que el anterior módulo, que se mencionan únicamente los dispositivos y elementos que entran dentro del diseño y que no han sido provistos por el INAMHI. En este sentido, no se toman en cuenta dentro de las especificaciones de costos: el sensor de precipitación, panel solar y controlador de carga.

#### 4.5.3. COSTO REFERENCIAL DEL MÓDULO DE ADQUISICIÓN DE DATOS

El costo referencial de este módulo se observa en la Tabla 4.8, en la cual se debe considerar que la PC utilizada como adquirente de datos es provista por el INAMHI, por tanto solo se considera en el costo referencial el dispositivo coordinador ZigBee.

**Tabla 4.8 Especificaciones de Costos de Elementos y Dispositivos del Módulo de Adquisición de Datos**

Descripción	Cantidad	Valor Unitario \$	Valor Total \$
Kit de desarrollo XBee-PRO 802.15.4 Starter Development Kit w/ 2 XBee-PRO Modules	1	315.18	315.18
<b>Total Costo Módulo de Adquisición de Datos</b>			<b>\$315.18</b>

*Fuente: Elaboración propia basado en facturas y proformas*

#### 4.5.4. COSTO REFERENCIAL DEL MÓDULO DE TRANSMISIÓN Y PROCESAMIENTO DE DATOS GPRS Y DE SOFTWARE DEL PROTOTIPO

El módulo de transmisión y procesamiento de datos GPRS contiene un modem WAVECOM, una fuente de voltaje de 12V en la estación remota de adquisición de

datos meteorológicos. De igual forma, contiene un servidor localizado en la estación central del INAMHI localizada en Quito.

El servidor es encargado de recopilar la información enviada vía GPRS a través de un router inalámbrico. En este sentido, todos los dispositivos y equipos de este módulo son provistos por INAMHI y por tanto no presentan costos referenciales.

Los softwares utilizados para el desarrollo y diseño del prototipo son gratuitos, para el caso del AVR Studio 6 y X-CTU. El software LabView 2012, no es gratuito, sin embargo, las licencias y herramientas del mismo han sido adquiridas por INAMHI. Por ende, tampoco se presentan especificaciones de costos referentes a software.

#### 4.5.5. COSTO TOTALES REFERENCIAL DEL PROTOTIPO

El costo total del prototipo se determina en base al costo individual de cada uno de los módulos que conforman el mismo. Por ende, se presenta en la Tabla 4.9, las especificaciones del costo total del prototipo.

**Tabla 4.9 Especificaciones del Costo Total del Prototipo**

<b>Descripción</b>	<b>Valor \$</b>
Costo Módulo de Temperatura y Humedad Relativa	364.23
Costo del Módulo de Precipitación	362.48
Costo del Módulo de Adquisición de Datos	315.18
<b>Costo Total del Prototipo</b>	<b>\$1,041.89</b>

*Fuente: Elaboración propia basado en facturas y proforma*

## CAPÍTULO 5

### CONCLUSIONES Y RECOMENDACIONES

#### 5.1. CONCLUSIONES

Las conclusiones se definen de acuerdo a los resultados y pruebas obtenidos en el proceso de diseño e implementación del prototipo, tomando en cuenta que la realimentación de información, cuando se cometía errores o se observaba que el diseño podría ser mejorado, es fundamental para optimizar el prototipo.

##### 5.1.1. CONCLUSIONES DE IMPLEMENTACIÓN, DISEÑO, VENTAJAS Y DESVENTAJAS DEL PROTOTIPO

- Tanto el diseño, como la implantación del prototipo, cumplen con el objetivo planteado referente a la implementación de un sistema inalámbrico orientado al monitoreo meteorológico, haciendo uso del estándar IEEE 802.15.4 y ZigBee que permite establecer un enlace en la banda libre ISM de 2.4GHz entre un módulo coordinador y dos módulos terminales que adquieren la información de temperatura, humedad y precipitación para su correspondiente procesamiento y almacenamiento, de tal manera que el INAMHI obtiene la información meteorológica de estos tres parámetros a través de la red GPRS.
- A nivel de hardware las protecciones que este prototipo presenta son: el controlador de carga para sobrecargas por parte del panel solar, circuito de protección de corriente “Foldback”, localizado en el circuito de detección de nivel de voltaje conectado directamente al canal AD1 del microprocesador y diodos de protección localizados en los reguladores de voltaje de 5V.
- El diseño e implementación de este prototipo establecen un bajo consumo de energía, bajo costo y el procesamiento de los datos meteorológicos no necesitan altas velocidades de transmisión. Además, los sensores pueden ser ubicados en puntos de medición independientes de la ubicación del adquisidor de datos, siempre y cuando se encuentren dentro del área de cobertura de la red inalámbrica ZigBee.

- Las aplicaciones que este prototipo puede generar, es el del correspondiente tratamiento de la información de temperatura, humedad y precipitación, con el fin de implementar modelos meteorológicos que permitan predecir el clima.
- El impacto que este prototipo podría tener a futuro, es la utilización e implementación de redes inalámbricas para la correspondiente adquisición y procesamiento de variables hidrometeorológicas.
- El costo del prototipo es de \$1,166.92, sin tener en cuenta el costo de algunos elementos que han sido provistos por el INAMHI, ni el costo de conocimiento y mano de obra. En este sentido el costo real aumentaría, tomando en cuenta que he estimado un costo de conocimiento y mano de obra en \$1000.
- Las limitaciones que presenta el prototipo son: rango de cobertura limitado y establecido por estándares IEEE 802.15.4 y ZigBee, muestreo de la información cada cinco minutos por parte del servidor GPRS, la solución indoor para el módulo de adquisición de datos implementada necesariamente necesita de energía AC y únicamente se maneja información de temperatura, humedad y precipitación, tomando en cuenta que las variables meteorológicas es bastante amplia.

#### **5.1.2. CONCLUSIONES DE PRUEBAS Y RESULTADOS DEL PROTOTIPO**

- Se concluye que el funcionamiento del prototipo es óptimo, luego de haber verificado la correcta ejecución del procesamiento, adquisición, recepción y transmisión de datos de todos los módulos y sistemas que conforman al mismo, durante 3 meses, tiempo en el cual se realizaron las pruebas y resultados del prototipo.
- De los resultados obtenidos de un conjunto de muestras registradas durante dos días consecutivos, de la comparación de valores de temperatura adquiridos por el prototipo y una estación de referencia del INAMHI, se observa que un 95.4% de los datos de temperatura son realmente efectivos y pueden ser utilizados por el INAMHI, ya que se encuentran dentro del rango de tolerancia estimado en:  $x \pm 0.4^{\circ}\text{C}$ . El 4.6% de datos restantes deben ser descartados ya que sobrepasan el umbral de tolerancia. Esto significa, que de las 2880 muestras registradas, aproximadamente 2753 son efectivas y 127 deben ser descartadas. Por lo tanto, se concluye que la adquisición de datos

de temperatura es óptima en un 95.4%. Este valor debe incrementarse hasta obtener 1% de datos descartados, mediante el rediseño y ajustes tanto en el circuito interfaz de transductor, como en el programa de conversión ADC, que son secciones donde se detectaron errores luego de haber realizado pruebas y haber obtenido los correspondientes resultados de temperatura.

- Los resultados obtenidos de un conjunto de muestras registradas durante dos días consecutivos, de la comparación de valores de humedad adquiridos por el prototipo y una estación de referencia del INAMHI, muestran que un 92.1% de los datos de temperatura son realmente efectivos y pueden ser utilizados por el INAMHI, ya que se encuentran dentro del rango de tolerancia estimado en:  $x \pm 1.4\%$ . El 7.9% de datos restantes deben ser descartados ya que sobrepasan el umbral de tolerancia. Esto significa, que de las 2880 muestras registradas, aproximadamente 2652 son efectivas y 228 deben ser descartadas. Por lo tanto, se concluye que la adquisición de datos de humedad es óptima en un 92.1%. Este valor debe incrementarse hasta obtener 1% de datos descartados, mediante el rediseño y ajustes tanto en el circuito interfaz de transductor, como en el programa de conversión ADC, que son secciones donde se detectaron errores luego de haber realizado pruebas y haber obtenido los correspondientes resultados de humedad.
- En referencia al módulo de precipitación, se cumple con los procesos de calibración del sensor, luego de haber obtenido un conteo de 21 basculaciones, al haber vertido 99ml de agua en el sensor acondicionado con un embudo de calibración, tal como se estipula en los procesos de calibración de precipitación establecidos por el laboratorio de meteorología del INAMHI.
- Finalmente, se concluye que a pesar de que el prototipo funcione correctamente en cuanto a la adquisición, recepción y transmisión de información meteorológica, se debe mejorar el procesamiento de la información hasta obtener un 99% de datos efectivos que pueden ser utilizados por el INAMHI.

## 5.2. RECOMENDACIONES

- Ampliar el alcance de la red inalámbrica ZigBee mediante la utilización de dispositivos ruteadores, los cuales además, permiten la comunicación con otras redes ZigBee.
- Incrementar el número de sensores y dispositivos finales que permitan una mayor cobertura de parámetros meteorológicos.
- Aumentar el tipo de tramas API en el programa de Tx/Rx serial, para incrementar el número de funcionalidades de gestión y control que permite el estándar ZigBee.
- Implementar mejores técnicas de corrección de errores en transmisión como en recepción, para establecer un prototipo más robusto.
- Implementar un método de encriptación de datos en ambas redes inalámbricas.
- A lo largo del diseño e implementación del prototipo se detectaron fallas en la conversión ADC de las señales de voltaje de temperatura y humedad. En ese sentido se recomienda tal como se mostró en las conclusiones de resultados y pruebas, el rediseño y ajustes tanto en el circuito interfaz de transductor, como en el programa de conversión ADC, para que el voltaje de referencia sea exacto, ya que las variaciones del mismo presentan errores en la conversión.
- Se presentaron fallas en el circuito optoacoplador, sobre todo cuando el mismo fue implementado en la placa PCB. Se recomienda implementar otro tipo de circuito que permita la detección de pulsos de forma más precisa.
- La presentación de información meteorológica tanto en el adquisidor de datos como en el servidor GPRS, fue complicado de diseñar y presenta limitaciones. Se recomienda la utilización de otro software que permita el manejo y control de base datos que sea más sencillo y efectivo.
- Cambiar los módulos XBee-PRO por los nuevos módulos XBee-PRO ZB para un mayor rango de cobertura.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] «ZigBee Specification,» [En línea]. Available: [www.zigbee.org](http://www.zigbee.org).
- [2] S. Farahani, «ZigBee Basics,» de ZigBee Wireless Networks and Transceivers, Burlington, Elsevier, 2008.
- [3] G. J., «Low-Rate Wireless Personal Area Networks,» IEEE Press, 2007.
- [4] P. Hidalgo, Folleto Telemática, Quito, Marzo 2010.
- [5] S. Farahani, «Protocolos de Capas ZigBee e IEEE 802.15.4,» de ZigBee Wireless Networks and Transceivers, Burlington, Elsevier, 2008.
- [6] G. Sanders, L. Thorens, M. Reisky, O. Rulik y S. Deylitz, «The General Packet Radio Service,» de GPRS Networks, England, Wiley, 2003.
- [7] G. Sanders, L. Thorens, M. Reisky, O. Rulik y S. Deylitz, «Core Network,» de GPRS Networks, England, Wiley, 2003.
- [8] G. Sanders, L. Thorens, M. Reisky, O. Rulik y S. Deylitz, «Interfaces and Protocols,» de GPRS Networks, England, Wiley, 2003.
- [9] W. Stallings, «Protocolos de Transporte,» de Comunicaciones y Redes de Computadores, Madrid, Prentice Hall, 2000.
- [10] G. Sanders, L. Thorens, M. Reisky, O. Rulik y S. Deylitz, «Towards All-IP Networks,» de GPRS Networks, England, Wiley, 2003.
- [11] «Temas de Interes,» 05 Octubre 2012. [En línea]. Available: [www.inamhi.gob.ec](http://www.inamhi.gob.ec).
- [12] INAMHI Dirección de Planificación, «Proyecto Fortalecimiento del INAMHI en Apoyo a la Gestión Integral del Riesgo de Desastres Naturales y del Cambio Climático en Ecuador,» Quito, 2010.
- [13] Instituto de Hidrología, Meteorología y Estudios Ambientales IDEAM, Manual del Observador Meteorológico, Medellín, 2001.
- [14] Campbell Scientific, «Partes de una Estación Meteorológica,» [En línea]. Available:[ftp://ftp.campbellsci.co.uk/pub/csl/outgoing/es/leaflets/estaciones\\_meteo\\_campbell.pdf](ftp://ftp.campbellsci.co.uk/pub/csl/outgoing/es/leaflets/estaciones_meteo_campbell.pdf).
- [15] L. Ing. Cobacango y R. Ing. Escobar, «Informe Técnico No IDT-01-2012 Actividades realizadas para Operación y Mantenimiento de la Red de Estaciones Hidrometeorológicas Automáticas durante el 2011,» Quito, 2012.

- [16] Vaisala, «HMP45A y HMP45D Relative Humidity and Temperature Probes,» Woburn.
- [17] Atmel, Octubre 2012. [En línea]. Available: [www.atmel.com](http://www.atmel.com).
- [18] Digi International Inc., Octubre 2012. [En línea]. Available: [www.digi.com](http://www.digi.com).
- [19] Texas Electronics, Inc., «Series 525 Rainfall Sensors Users Manual,» Dallas.
- [20] Advantech, «Industrial Automation,» 15 Octubre 2012. [En línea]. Available: [http://www.advantech.com/products/TPC-1071H/mod\\_408CF426-12BC-4C7B-BE5B-D93591CA06E4.aspx](http://www.advantech.com/products/TPC-1071H/mod_408CF426-12BC-4C7B-BE5B-D93591CA06E4.aspx).
- [21] «Product Details,» 15 Octubre 2012. [En línea]. Available: [http://www.wavecom.com.au/product\\_view.php?id\\_product=432](http://www.wavecom.com.au/product_view.php?id_product=432).
- [22] S. F. Barrett y D. J. Pack, «Analog to Digital Conversion,» de Atmel AVR Microcontroller Primer: Programming and Interfacing, Southern Methodist University, Mitchell A. Thornton, 2008.
- [23] «Amplificadores Operacionales,» de Teoría de Circuitos y Dispositivos Electrónicos, México, Prentice Hall, 2003.
- [24] Atmel International Inc., «Product Manual 8-bit AVR Microcontroller with 16K/32K/64K Bytes In-System Programmable Flash,» 2010.
- [25] «Timing Subsystem,» de Atmel AVR Microcontroller Primer: Programming and Interfacing, Southern Methodist University, Mitchell A. Thornton, 2008.
- [26] Hitachi, Datasheet LM044L: LCD de 4x20 Caracteres.
- [27] O. Buitron, Apuntes Cuaderno Sistemas Microprocesados, Quito, Pichincha, 2009.
- [28] Digi International Inc., «Product Manual XBee/XBee-PRO OEM RF Modules,» Minnetonka, 2008.
- [29] CONELEC (Consejo Nacional de Electricidad), «Atlas Solar del Ecuador con Fines de Generación Eléctrica,» Quito, 2008.
- [30] ProViento S.A., 2011. [En línea]. Available: <http://www.proviento.com.ec/>.
- [31] CSB Battery Co., 2011. [En línea]. Available: <http://www.alldatasheet.es/datasheet-pdf/pdf/273233/CSB/BATTERY/GPL12750.html>.
- [32] SunSaver, 2011. [En línea]. Available: <http://www.solarpanelstore.com/solar-power.small-charge-controllers.sunsaver.sunsaver-6-12.info.1.html>.

- [33] Texas Instruments, 2003. [En línea]. Available: <http://www.sparkfun.com/datasheets/Components/LM7805.pdf>.
- [34] SparkFun Electronics, 2012. [En línea]. Available: <https://www.sparkfun.com/products/9132>.
- [35] J. M. Drake Moyan, «Instrumentación Electrónica de Comunicaciones: Referencias de Tensión y Reguladores de Tensión.,» Santander, 2005.
- [36] National Instruments Corporation., LabVIEW Basics II: Development Course Manual, National Instruments Corporation, 2004.
- [37] National Instruments Corporation., LabView Basics I: Introduction Course Manual, National Instruments Corporation., 2004.
- [38] «Modelo de Interacción Cliente-Servidor,» de Redes Globales de Información con Internet y TCP/IP, Juarez, Prentice-Hall, 1996.
- [39] Wavecom, AT Commands Interface Guide for Open AT Firmware v7.4, 2009.
- [40] IDT-INAMHI, «Manual de Ingeniería Modem Wavecom,» Quito, 2008.
- [41] Fis. Wladimir Eduardo, Banda Barragán; INAMHI, «Manual de Confirmación Metrológica en Laboratorio: Capítulo 1 Calibración de Temperatura,» Quito, 2011.
- [42] W. E. Fís. Banda Barragán, «Manual de Confirmación Meteorológica en Laboratorio: Capítulo 3 Calibración de Precipitación,» Quito, 2011.
- [43] «Changes in the Radio Subsystem for GPRS,» de GPRS Networks, England, Wiley, 2003.
- [44] L. Ing. Cobacango, «Procedimiento IDT0006 Instalación AWS Logotronic,» Quito, 2010.
- [45] L. Ing. Cobacango, «Procedimiento IDT0011 Instalación Vaisala MAWS301,» Quito, 2011.
- [46] C. Pastor Pérez, «Análisis (químico) y control (digital) en la producción del vino,» Castilla-La Mancha, 2003.
- [47] P. López González, «Desarrollo de un nodo sensor para la monitorización climática/ambiental.,» Catalunya, 2008.
- [48] Wavecom, «Fastrack Supreme User Guide,» 2008.

## ANEXOS

### A. PROGRAMAS DE PROCESAMIENTO DE INFORMACIÓN DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA

#### A.1. PROGRAMA PRINCIPAL DEL MÓDULO DE TEMPERATURA Y HUMEDAD RELATIVA

```
/* ////////////////////////////////////////
 *
 * NOMBRE DEL PROYECTO: Adqusicion_Temperat.cpp
 * FUNCIÓN: Programa principal que permite adquirir y procesar los
 *         valores de voltaje del sensor de temperatura y humedad,
 *         para su posterior envío al correspondiente dispositivo
 *         ZigBee. Adicionalmente los valores de temperatura y
 *         humedad obtenidos en el puerto serial, serán presentados
 *         en el LCD.
 * FECHA DE CREACIÓN: 11/06/2013
 * AUTOR: Miguel Angel Bonilla Vergara.
 *
 * ////////////////////////////////////////

//*****/
//* 1) DEFINICIÓN DE LIBRERÍAS.
//*****/

/* Habilitación de la librería para el manejo de las entradas/salidas
   digitales.*/

#include <avr/io.h>

/* Habilitación de la librería para la conversión de cadenas de
   caracteres y valores numéricos.*/

#include <stdlib.h>

/* Habilitación de la librería para operaciones matemáticas básicas.*/

#include <math.h>

/* Habilitación de la librería para operaciones con cadenas de
   caracteres.*/

#include <string.h>

/* Habilitación de la librería para el manejo de variables tipo
   enteras.*/

#include <stdint.h>

/* Habilitación de la librería para el manejo de subrutinas de
   interrupciones.*/
```

```

#include <avr/interrupt.h>

/* Definición y establecimiento de la frecuencia del oscilador
externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL
#endif

/* Habilidad de la librería para el manejo de retardos.*/

#include <avr/delay.h>

//*****
/* 2) DEFINICIÓN DE LIBRERIAS PREVIAMENTE ESTRUCTURADAS.
//*****

/* Habilidad de las librerías para el manejo de los siguientes
funciones:
*
* - Conversión ADC simple.
* - Manejo del LCD.
* - Muestreo de la conversión ADC.
* - Manejo de interrupciones externas.
* - Manejo del interfaz UART.
*
*/

#include "adc.h"
#include "lcd.h"
#include "muestreo.h"
#include "interrupcion_externa.h"
#include "uart.h"

//*****
/* 3) DECLARACIÓN DE VARIABLES GLOBALES.
//*****

/* Las variables globales a ser utilizadas tienen como tarea:
*
* - Obtención de la cadena de caracteres de los valores de voltaje
de los sensores de temperatura y humedad.
* - Obtención de la cadena de caracteres de los valores de
temperatura y humedad recibidos mediante el interfaz UART.
* - Obtención de la cadena de caracteres que indica el porcentaje
del nivel de batería de funcionamiento del circuito.
*
*/

char caracteres_canal_ADC_obtenidos[7];
char Caracteres_Recibidos[10];
char *caracteres_nivel_bateria_obtenidos;

//*****
/* 4) DECLARACIÓN DE FUNCIONES.
//*****

/* Las funciones que se definen a continuación presentan las
siguientes tareas:
*

```

```

* - Inicialización y configuración de los pórtricos de entrada
* y salida digitales de proposito general.
* - Inicialización de las interrupciones de los subsistemas
* que se utilizarán durante la ejecución del programa.
* - Lectura de los valores de voltaje del sensor de temperatura.
* - Lectura de los valores de voltaje del sensor de humedad.
* - Lectura del porcentaje de batería para funcionamiento del
* circuito.
*
*/

void inic_porticos(void);
void inic_interrupciones(void);
char * lectura_voltaje_sensor_temperatura(void);
char * lectura_voltaje_sensor_humedad(void);
char * lectura_nivel_bateria(void);

//*****
/* 5) PROGRAMA PRINCIPAL.
//*****

int main(void)
{
    /* Llamado y ejecución de funciones previamente declaradas y
    correspondientes a:
    *
    * - Inicialización de pórtricos de entradas/salidas digitales.
    * - Inicialización del manejo del LCD.
    * - Inicialización de la conversión ADC simple.
    * - Inicialización del temporizador contador 1.
    * - Inicialización del interfaz UART.
    * - Inicialización de las interrupciones externas.
    *
    */

    inic_porticos();
    inic_LCD();
    inic_ADC_simple();
    inic_timer_counter_1();
    inic_USART();
    inic_interrupciones();

    /* Encendido de la pantalla del LCD.*/

    PORTA |= (1 << PA4);

    /* Declaración de variables locales las cuales son utilizadas
    * para:
    *
    * - Presentación inicial a ser mostrada en el LCD.
    * - Lectura de caracteres de voltaje a ser enviados.
    * - Lectura de los caracteres recibidos.
    *
    */

    char linea1[] = "EPN-INAMHI ";
    char linea3[] = "MOD. TEMP. HUMED.";
    char linea4[] = "MIGUEL BONILLA";
    char *caracteres_voltajes_obtenidos;
    char *caracteres_recibidos;

```

```

/* Ejecución de comandos que son enviados al LCD para
   activación del cursor, borrado de pantalla y ubicación
   del cursor en posición inicial.*/

comandos_LCD(0x0E);
comandos_LCD(Borrar_Pantalla);
comandos_LCD(Posicion_Inicial);

/* Posicionamiento del cursor en la correspondiente fila Y
   columna para la correspondiente escritura de los caracteres
   previamente definidos.*/

posicionar_cursor(1,6);
escribir_cadena_caracteres_LCD(linea1);
posicionar_cursor(3,2);
escribir_cadena_caracteres_LCD(linea3);
posicionar_cursor(4,4);
escribir_cadena_caracteres_LCD(linea4);

/* Inicialización del contador de pulsos.*/

contador_pulsos = 0;

/* Ejecución de un retardo correspondiente a 9 segundos.*/

_delay_ms(9000);

/* Desactivación del LCD y apagado de la pantalla.*/

comandos_LCD(0x0A);
PORTA &= ~(1 << PA4);

/* Ejecución de un lazo infinito.*/

while(1)
{
    _delay_ms(50);

    /* Lectura del voltaje del sensor de temperatura para
       su correspondiente transmisión mediante el uso del
       interfaz USART.*/

    caracteres_voltajes_obtenidos = lectura_voltaje_sensor_temperatura();

    enviar_datos_temperatura_al_buffer_TX(caracteres_voltajes_obtenidos);

    /* Lectura del voltaje del sensor de humedad para
       su correspondiente transmisión mediante el uso del
       interfaz USART.*/

    caracteres_voltajes_obtenidos = lectura_voltaje_sensor_humedad();
    enviar_datos_humedad_al_buffer_TX(caracteres_voltajes_obtenidos);

    /* Lectura del porcentaje del nivel de batería para
       su correspondiente transmisión mediante el uso del
       interfaz USART.*/

    caracteres_nivel_bateria_obtenidos = lectura_nivel_bateria();

    enviar_datos_nivel_bateria_al_buffer_TX(caracteres_nivel_bateria_obtenidos);
}

```

```

        /* Presentación del nivel de batería en el LCD.*/

        posicionar_cursor(4,12);
        escribir_cadena_caracteres_LCD(caracteres_nivel_bateria_obtenidos);

        /* Retardo de 50ms.*/

        _delay_ms(50);

        /* Recepción de las tramas de datos que empiecen con
           el caracter T, y escritura del valor de temperatura
           procesado en el LCD.*/

        caracteres_recibidos = recibir_datos_buffer_RX('T');
        posicionar_cursor(2,12);
        escribir_cadena_caracteres_LCD(caracteres_recibidos);

        /* Retardo de 50ms.*/

        _delay_ms(50);

        /* Recepción de las tramas de datos que empiecen con
           el caracter H, y escritura del valor de humedad
           procesado en el LCD.*/

        caracteres_recibidos = recibir_datos_buffer_RX('H');
        posicionar_cursor(3,12);
        escribir_cadena_caracteres_LCD(caracteres_recibidos);

        /* Verificación del conteo de segundos correspondiente a
           5 minutos, luego de los cuales se procede a desactivar
           el LCD y apagar la pantalla.*/

        if (segundos_lcd == 300)
        {
            comandos_LCD(0x0A);
            PORTA &= ~(1 << PA4);
        }
    }
}

//*****
/** 6) FUNCIÓN DE INICIALIZACIÓN DE PÓRTICOS.
//*****

void inic_porticos(void)
{
    /* Habilitación de la salida de datos del LCD.*/

    DDRC = 0xFF;
    PORTC = 0x00;

    /* Habilitación de la salida de datos RS, RW y E del LCD.*/

    DDRA |= (1<<PA5) | (1<<PA6) | (1<<PA7);
    PORTA &= ~(1<<PA5) | (1<<PA6) | (1<<PA7));

    /* Habilitación de los canales de conversión ADC: AD0, AD1 y
       AD2 como entradas digitales.*/

```

```

DDRA  &= ~(1<<PA0);
PORTA &= ~(1<<PA0);

DDRA  &= ~(1<<PA1);
PORTA &= ~(1<<PA1);

DDRA  &= ~(1<<PA2);
PORTA &= ~(1<<PA2);

/* Habilitación del pin OC1A como salida digital.*/

DDRD |= (1<<PD5);
PORTD &= ~(1<<PD5);

/* Habilitación del pin OC0A como salida digital.*/

DDRB |= (1<<PB3);
PORTB &= ~(1<<PB3);

/* Habilitación de la interrupción externa INT0 como
   entrada digital de datos.*/

DDRD &= ~(1<<PD2);
PORTD &= ~(1<<PD2);

/* Habilitación de la interrupción externa INT1 como
   entrada digital de datos.*/

DDRD &= ~(1<<PD3);
PORTD &= ~(1<<PD3);

/* Habilitación del pin para encendido o apagado del LCD
   como salida digital de datos.*/

DDRA |= (1<<PA4);
PORTA &= ~(1<<PA4);

/* Habilitación de la recepción USART0 como entrada
   digital de datos.*/

DDRD &= ~(1<<PD0);
PORTD &= ~(1<<PD0);

/* Habilitación de la transmisión USART0 como salida
   digital de datos.*/

DDRD |= (1<<PD1);
PORTD &= ~(1<<PD1);

/* Activación Pull-ups.*/

DDRD &= ~((1<<PD4) | (1<<PD6) | (1<<PD7));
PORTD |= (1<<PD4) | (1<<PD6) | (1<<PD7);

DDRA &= ~(1<<PA3);
PORTA |= (1<<PA3);

DDRB &= ~((1<<PB0) | (1<<PB1) | (1<<PB2) | (1<<PB4) | (1<<PB5) | (1<<PB6) |
          (1<<PB7));
PORTB |= (1<<PB0) | (1<<PB1) | (1<<PB2) | (1<<PB4) | (1<<PB5) | (1<<PB6) |
          (1<<PB7);

```

```

}

//*****
/* 7) FUNCIÓN DE INICIALIZACIÓN DE INTERRUPTIONES.
//*****

void inic_interrupciones(void)
{
    /* Habilitación de las interrupciones globales.*/

    sei();

    /* Habilitación de la interrupción de conversión ADC simple.*/

    ADCSRA |= (1<<ADIE);

    /* Habilitación de la interrupción cuando se presenta la
       detección de comparación en OCR1B.*/

    TIMSK1 |= (1<<OCIE1B);

    /* Habilitación de la interrupción cuando se presenta la
       detección de comparación en OCR1A.*/

    TIMSK1 |= (1<<OCIE1A);

    /* Habilitación de la interrupción de recepción completa del
       interfaz USART0.*/

    UCSR0B |= (1<<RXCIE0);

    /* Habilitación de la interrupción externa INT1, la cual se
       activa con la detección de una transición positiva.*/

    EIMSK |= (1<<INT1);
    EICRA |= (1<<ISC10) | (1<<ISC11);
}

//*****
/* 8) FUNCIÓN DE LECTURA DEL VOLTAJE DEL SENSOR DE TEMPERATURA.
//*****

char * lectura_voltaje_sensor_temperatura(void)
{
    /* Declaración de variables locales.*/

    float voltaje_canal_ADC_flotante = 0;
    int voltaje_canal_ADC_entero = 0;

    /* Lectura del voltaje del canal AD0.*/

    voltaje_canal_ADC_flotante = lectura_canal_ADC(0);

    /* Conversión del valor flotante de voltaje del canal AD0, en
       un valor entero.*/

    voltaje_canal_ADC_entero = (int)(voltaje_canal_ADC_flotante);

    /* Conversión del valor entero de voltaje del canal AD0, en una
       cadena de caracteres.*/

```

```

        itoa( voltaje_canal_ADC_entero, caracteres_canal_ADC_obtenidos, 10 );

        /* Retorno de la cadena de caracteres correspondiente a la lectura
           del voltaje del canal AD0, al programa principal.*/

        return caracteres_canal_ADC_obtenidos;
    }

    /*******
    /*** 9) FUNCIÓN DE LECTURA DEL VOLTAJE DEL SENSOR DE HUMEDAD.
    /*******

char * lectura_voltaje_sensor_humedad(void)
{
    /* Declaración de variables locales.*/

    float voltaje_canal_ADC_flotante = 0;
    int voltaje_canal_ADC_entero = 0;

    /* Lectura del voltaje del canal AD2.*/

    voltaje_canal_ADC_flotante = lectura_canal_ADC(2);

    /* Conversión del valor flotante de voltaje del canal AD2, en
       un valor entero.*/

    voltaje_canal_ADC_entero = (int)(voltaje_canal_ADC_flotante);

    /* Conversión del valor entero de voltaje del canal AD2, en una
       cadena de caracteres.*/

    itoa( voltaje_canal_ADC_entero, caracteres_canal_ADC_obtenidos, 10 );

    /* Retorno de la cadena de caracteres correspondiente a la lectura
       del voltaje del canal AD2, al programa principal.*/

    return caracteres_canal_ADC_obtenidos;
}

    /*******
    /*** 10) FUNCIÓN DE LECTURA DEL NIVEL DE VOLTAJE DE LA BATERÍA A.
    /*******

char * lectura_nivel_bateria(void)
{
    /* Declaración de variables locales.*/

    float Nivel_Bateria = 0;
    float voltaje_canal_ADC = 0;

    /* Lectura del voltaje del canal AD1.*/

    voltaje_canal_ADC = lectura_canal_ADC(1);

    /* Comprobación de que el nivel de voltaje sea mayor a 1.3,
       el cual representa el voltaje mínimo de operación.*/

    if (voltaje_canal_ADC > 1.3)
    {
        /* Determinar el voltaje de entrada al conversor ADC,
           tomando en cuenta que el rango de operación se

```

```

        encuentra entre 2.5V y 1.3V.*/
    voltaje_canal_ADC = voltaje_canal_ADC - 1.3;
}
else
{
    /* En este caso el voltaje de operación es menor a
    al valor mínimo, por lo cual el porcentaje debe
    ser igual a cero.*/

    voltaje_canal_ADC = 0;
}

/* Determinación del nivel de batería en porcentaje.*/
Nivel_Bateria = (voltaje_canal_ADC * 100)/1.26;

if (Nivel_Bateria > 100) Nivel_Bateria = 100;

/* Conversión del nivel de batería en una cadena de caracteres.*/

dtostrf( Nivel_Bateria, 3, 0, caracteres_canal_ADC_obtenidos );

/* Retorno de la cadena de caracteres del nivel de batería al
programa principal.*/

return caracteres_canal_ADC_obtenidos;
}

```

## A.2. PROGRAMA DE CONVERSIÓN ANALÓGICO A DIGITAL, CORRESPONDIENTE A LA LECTURA DE TRES CANALES AD DEL MICROPROCESADOR

```

/* ////////////////////////////////////////
*
* NOMBRE DEL PROYECTO: adc.h
* FUNCIÓN: Librería que permite realizar la conversión ADC simple
*          en los canales AD0, AD1 y AD2 en orden secuencial.
* FECHA DE CREACIÓN: 24/11/2013
* AUTOR: Miguel Angel Bonilla Vergara.
*
* ////////////////////////////////////////

//*****/
/* 1) DEFINICIÓN DE LIBRERÍAS.
//*****/

/* Definición de la librería adc.h a ser utilizado en el programa
principal.*/

#ifndef ADC_H_
#define ADC_H_

/* Habilitación de la librería para el manejo de las entradas/salidas
digitales.*/

```

```

#include <avr/io.h>

/* Habilitación de la librería para el manejo de variables tipo
   enteras.*/

#include <stdint.h>

/* Habilitación de la librería para la conversión de cadenas de
   caracteres y valores numéricos.*/

#include <stdlib.h>

/* Habilitación de la librería para operaciones con cadenas de
   caracteres.*/

#include <string.h>

/* Habilitación de la librería para el manejo de subrutinas de
   interrupciones.*/

#include <avr/interrupt.h>

/* Definición y establecimiento de la frecuencia del oscilador
   externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL /* cristal de 4MHz */
#endif

/* Habilitación de la librería para el manejo de retardos.*/

#include <avr/delay.h>

//*****
/* 2) DECLARACIÓN DE VARIABLES GLOBALES.
//*****

/* Las variables globales a ser utilizadas tienen como tarea:
*
* - Establecer una bandera ADC que indica la conversión del
* siguiente canal ADC.
* - Obtención del valor flotante del canal AD0.
* - Obtención del valor flotante del canal AD1.
* - Obtención del valor flotante del canal AD2.
*
*/

uint8_t bandera_canal_ADC = 0;
float voltaje_AD0 = 0;
float voltaje_AD1 = 0 ;
float voltaje_AD2 = 0 ;

//*****
/* 3) DECLARACIÓN DE FUNCIONES.
//*****

/* Las funciones que se definen a continuación presentan las
* siguientes tareas:
*
* - Configuración de la conversión ADC simple.

```

```

* - Lectura del valor del canal ADC.
*
*/

void inic_ADC_simple(void);
float lectura_canal_ADC(unsigned char canal_ADC);

//*****
/* 4) FUNCIÓN DE CONFIGURACIÓN DEL SUBSISTEMA ADC DEL AVR.
//*****

void inic_ADC_simple (void)
{
    /* Configuración de la Conversión ADC Simple.*/

    /* La configuración de los registros bit por bit del ADC,
    * permite que se realice las siguientes funciones:
    *
    * - Habilitación del ADC.
    * - Habilitación del modo de auto-conversión con una fuente
    *   FREE-RUNNING.
    * - Establecimiento de un valor de preescalamiento de 8 para
    *   obtener una frecuencia de reloj para operación del ADC (clkADC)
    *   entre 50KHz y 200kHz.
    * - Establecimiento de un voltaje de referencia interno de 2.56V.
    * - Habilitación del ajuste de resultado en la derecha.
    * - Habilitación del canal 0 (AD0) para el ingreso de una señal
    *   analógica.
    * - Des-habilitación de los canales ADC que no se van a utilizar.
    *
    */

    ADCSRA = (1<<ADEN) | (1<<ADATE) | (0<<ADIE) | (1<<ADPS2) | (0<<ADPS1) |
              (1<<ADPS0);
    ADCSRB = (1<<ADTS2) | (0<<ADTS1) | (1<<ADTS0);
    ADMUX   = (0<<REFS1) | (1<<REFS0) | (0<<ADLAR);

    /* Habilitación del canal AD0 que inicialmente se va a leer.*/

    ADMUX  &= ~(1<<MUX4);
    ADMUX  &= ~(1<<MUX3);
    ADMUX  &= ~(1<<MUX2);
    ADMUX  &= ~(1<<MUX1);
    ADMUX  &= ~(1<<MUX0);

    /* Desactivación de las entradas digitales de los canales en los
    que se realice la conversión ADC.*/

    DIDR0 = (0<<ADC7D) | (0<<ADC6D) | (0<<ADC5D) | (0<<ADC4D) | (0<<ADC3D) |
            (1<<ADC2D) | (1<<ADC1D) | (1<<ADC0D);
}

//*****
/* 5) FUNCIÓN DE LECTURA DE LOS CANALES ADC.
//*****

float lectura_canal_ADC(unsigned char canal_ADC)
{
    /* Declaración de variables locales.*/

    float voltaje_ADC = 0;

```

```

    /* Lectura del canal AD0.*/

    if (canal_ADC == 0)
    {
        voltaje_ADC = voltaje_AD0;
    }

    /* Lectura del canal AD1.*/

    if (canal_ADC == 1)
    {
        voltaje_ADC = voltaje_AD1;
    }

    /* Lectura del canal AD2.*/

    if (canal_ADC == 2)
    {
        voltaje_ADC = voltaje_AD2;
    }

    /* Retorno del voltaje del canal ADC leído.*/

    return voltaje_ADC;
}

/*****
/* 6) FUNCIÓN DE INTERRUPCIÓN DE CONVERSIÓN ADC SIMPLE.
*****/

ISR(ADC_vect)
{
    /* Desactivación del subsistema ADC del AVR para tomar los
    valores de los registros ADC.*/

    ADCSRA &= ~(1 << ADEN);

    /* Declaración de variables locales para lectura de los
    registros correspondientes a los bits menos y más
    significativos, conversión ADC entero y flotante y
    el valor de voltaje de entrada flotante.*/

    unsigned int registro_ADCL = 0;
    unsigned int registro_ADCH = 0;
    unsigned int conversion_ADC_entero = 0;
    float conversion_ADC_flotante = 0;
    float vin_flotante = 0;

    /* Verificación de la bandera que indica que canal debe
    ser leído.*/

    if (bandera_canal_ADC == 0)
    {
        /* Habilitación del siguiente canal AD1 a ser leído.*/

        ADMUX &= ~(1<<MUX4);
        ADMUX &= ~(1<<MUX3);
        ADMUX &= ~(1<<MUX2);
        ADMUX &= ~(1<<MUX1);
        ADMUX |= (1<<MUX0);
    }
}

```

```

/* Inicialmente se procede con la lectura de los 8-bits
   del registro ADCL.*/

registro_ADCL = ADCL;

/* A continuación se procede con la lectura de los 8-bits
   del registro ADCH.*/

registro_ADCH = ((unsigned int)(ADCH << 8));

/* Concatenación de la lectura de ambos registros en una sola
   variable, y conversión de esta variable en un solo valor
   flotante.*/

conversion_ADC_entero = registro_ADCL | registro_ADCH;
conversion_ADC_flotante = (float)(conversion_ADC_entero);

/* Ecuación de conversión de un voltaje de entrada en el
   canal AD0, y su correspondiente almacenamiento en una
   variable local. */

vin_flotante = (conversion_ADC_flotante * 5000)/1024;
voltaje_AD0 = vin_flotante;

}

if (bandera_canal_ADC == 1)
{
    /* Habilitación del siguiente canal AD2 a ser leído.*/

    ADMUX  &= ~(1<<MUX4);
    ADMUX  &= ~(1<<MUX3);
    ADMUX  &= ~(1<<MUX2);
    ADMUX  |=  (1<<MUX1);
    ADMUX  &= ~(1<<MUX0);

    /* Inicialmente se procede con la lectura de los 8-bits
       del registro ADCL.*/

    registro_ADCL = ADCL;

    /* A continuación se procede con la lectura de los 8-bits
       del registro ADCH.*/

    registro_ADCH = ((unsigned int)(ADCH << 8));

    /* Concatenación de la lectura de ambos registros en una sola
       variable, y conversión de esta variable en un solo valor
       flotante.*/

    conversion_ADC_entero = registro_ADCL | registro_ADCH;
    conversion_ADC_flotante = (float)(conversion_ADC_entero);

    /* Ecuación de conversión de un voltaje de entrada en el
       canal AD1, y su correspondiente almacenamiento en una
       variable local. */

    vin_flotante = (conversion_ADC_flotante * 5)/1024;
    voltaje_AD1 = vin_flotante;
}

```



```

* FUNCIÓN: Librería que permite el manejo de un LCD de
*           4x20 caracteres.
* FECHA DE CREACIÓN: 24/11/2011
* AUTOR: Miguel Angel Bonilla Vergara.
*
*//////////////////////////////////////

//*****
//* 1) DEFINICIÓN DE LIBRERÍAS.
//*****

/* Definición de la librería uart.h a ser utilizado en el programa
principal.*/

#ifndef LCD_H_
#define LCD_H_

/* Habilitación de la librería para el manejo de las entradas/salidas
digitales.*/

#include <avr/io.h>

/* Definición y establecimiento de la frecuencia del oscilador
externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL
#endif

/* Habilitación de la librería para el manejo de retardos.*/

#include <avr/delay.h>

//*****
//* 2) DECLARACIÓN DE CONSTANTES Y VARIABLES GLOBALES.
//*****

/* Se define constantes que permite controlar los siguientes
* parámetros:
*
* - Bus de datos del LCD.
* - Bus de control del LCD.
* - Habilitación de escritura de comandos o datos en el LCD
*   el conbit RS.
* - Escritura en el LCD utilizando el bit RW.
* - Habilitación del LCD utilizando el bit E.
* - Definición del número de bits de datos.
*
*/

#define Bus_Datos PORTC
#define Bus_Control PORTA
#define RS PA5
#define RW PA6
#define E PA7
#define Numero_Filas 4
#define Numero_Columnas 20
unsigned int Numero_Bits_Datos = 8;

/* Comandos principales.*/

```

```

#define Borrarr_Pantalla    0x01
#define Posicion_Inicial    0x02

/* Se define los comandos para situar el cursor en el inicio
de cada línea.*/

#define DDRB_LINEA1 0x80 //1000 0000; posición inicial 0x00 de la DDRAM
#define DDRB_LINEA2 0xC0 //1100 0000; posición 0x40 de la DDRAM
#define DDRB_LINEA3 0x94 //1001 0100; posición 0x14 de la DDRAM
#define DDRB_LINEA4 0xD4 //1101 0100; posición 0x54 de la DDRAM

//*****
/* 3) DECLARACIÓN DE FUNCIONES.
//*****

/* Se declara funciones para cumplir con las siguientes tareas:
*
* - Inicialización del LCD.
* - Escritura de un solo caracter en el LCD.
* - Escritura de una cadena de caracteres en el LCD.
* - Posicionamiento del cursor del LCD.
* - Introducción de comandos en el LCD.
*
*/

void inic_LCD(void);
void escribir_caracter_LCD(unsigned char caracter);
void escribir_cadena_caracteres_LCD(char *cadena);
void posicionar_cursor(unsigned char x, unsigned char y);
void comandos_LCD(unsigned char comando);

//*****
/* 4) FUNCIÓN DE CONFIGURACIÓN DEL LCD.
//*****

void inic_LCD(void)
{
    /* Declaración de variables locales.*/

    unsigned int Bits_Datos = 0;

    _delay_ms(100);

    /* Esta función permite introducir comandos en el LCD que
    * permiten configurar:
    *
    * - El tipo de bus de datos del LCD.
    * - Activación del Display.
    * - Encendido o apagado del cursor.
    * - Encendido o apagado del parpadeo del cursor.
    * - Borrado de la pantalla.
    * - Ubicación del cursor en la posición 1 de la pantalla.
    * - Incremento de la posición del cursor.
    * - Desplazamiento o no del cursor.
    *
    */

    if (Numero_Bits_Datos == 4) Bits_Datos = 0x2C;
    if (Numero_Bits_Datos == 8) Bits_Datos = 0x3C;

    comandos_LCD(Bits_Datos);

```

```

    comandos_LCD(Bits_Datos);
    comandos_LCD(0x0C);
    comandos_LCD(Borrar_Pantalla);
    comandos_LCD(0x06);
}

/*****
/* 5) FUNCIÓN DE INTRODUCCIÓN DE COMANDOS EN EL LCD.
*****/

void comandos_LCD(unsigned char comando)
{
    /* Esta función permite realizar las siguientes tareas:
    *
    * - Escritura de comandos en el LCD mediante la activación del
    *   bit RS en modo de ingreso de comandos.
    * - Activación del bit RW en modo de escritura y habilitación.
    * - Desactivación del LCD durante el ingreso de los comandos
    *   en el LCD.
    *
    */

    Bus_Control &= ~(1<<RS);    //ingreso de comandos
    Bus_Control &= ~(1<<RW);    //escritura en el LCD
    Bus_Datos = comando;        //escritura del comando en el LCD
    Bus_Control |= (1<<E);      //habilitación del LCD
    _delay_us(100);
    Bus_Control &= ~(1<<E);    //des-habilitación del LCD
    _delay_ms(10);
}

/*****
/* 6) FUNCIÓN DE POSICIONAMIENTO DEL CURSOR EN EL LCD.
*****/

void posicionar_cursor(unsigned char x, unsigned char y) //línea x 1 a 4, carácter y
de 1 a 20.
{
    /* Esta función permite posicionar el cursor según la línea x
    y la columna y en el LCD. Se realiza entonces un barrido
    línea por línea, y dentro de cada línea realiza el barrido
    de las 20 columnas.*/

    unsigned char comando=DDRB_LINEA1;
    switch(x)
    {
        case 1:
        {
            comando=DDRB_LINEA1+y-1;
        }break;
        case 2:
        {
            comando=DDRB_LINEA2+y-1;
        }break;
        case 3:
        {
            comando=DDRB_LINEA3+y-1;
        }break;
        case 4:
        {
            comando=DDRB_LINEA4+y-1;
        }
    }
}

```

```

        }break;
        default:break;
    }
    /* Habilitación para escrituras de comandos en el LCD.*/

        comandos_LCD(comando);
}

/*****
/* 7) FUNCIÓN DE ESCRITURA DE CARACTERES INDIVIDUALES EN EL LCD.
*****/

void escribir_caracter_LCD(unsigned char caracter)
{
    /* Esta función permite la escritura de caracter por caracter en
    el LCD mediante la activación del bit RS en modo de ingreso de
    datos, activación del bit RW en modo de escritura y habilitación
    y desactivación del LCD durante la presentación de datos en el
    LCD.*/

    Bus_Control |= (1<<RS);
    Bus_Control &= ~(1<<RW);
    Bus_Datos = caracter;
    Bus_Control |= (1<<E);
    _delay_us(100);
    Bus_Control &= ~(1<<E);
    _delay_ms(10);
}

/*****
/* 8) FUNCIÓN DE ESCRITURA DE UNA CADENA DE CARACTERES EN EL LCD.
*****/

void escribir_cadena_caracteres_LCD(char *cadena)
{
    /* Esta función permite la escritura de una cadena de caracteres
    en el LCD mediante la activación del bit RS en modo de ingreso
    de datos, activación del bit RW en modo de escritura y
    habilitación y desactivación del LCD durante el ingreso de los
    datos en el LCD.*/

    char *inicio = cadena;
    int i=0;

    for(i=0;((inicio[i]!='\0')&&(i<Numero_Columnas));i++)
    {
        Bus_Control |= (1<<RS);    //RS0 RW0 caracter
        Bus_Control &= ~(1<<RW);
        Bus_Datos = inicio[i];
        Bus_Control |= (1<<E);
        _delay_us(100);
        Bus_Control &= ~(1<<E);
        _delay_ms(3);
    }
}

#endif /* LCD_H_ */

```

## A.4. PROGRAMA DE MUESTREO DE LA CONVERSIÓN ADC CADA SEGUNDO, POR MEDIO DE LA UTILIZACIÓN DE TEMPORIZADORES

```
/* ////////////////////////////////////////
*
* NOMBRE DEL PROYECTO: muestreo_ADC.h
* FUNCIÓN: Librería que permite realizar el muestreo de la conversión
*         ADC cada segundo, mediante la utilización
*         del Timer/Counter 1.
* FECHA DE CREACIÓN: 29/11/2011
* AUTOR: Miguel Angel Bonilla Vergara.
*
*//////////////////////////////////////

//*****
//* 1) DEFINICIÓN DE LIBRERÍAS.
//*****

/* Definición de la librería uart.h a ser utilizado en el programa
principal.*/

#ifndef MUESTREO_H_
#define MUESTREO_H_

/* Habilidad de la librería para el manejo de las entradas/salidas
digitales.*/

#include <avr/io.h>

/* Habilidad de la librería para el manejo de variables tipo
enteras.*/

#include <stdint.h>

/* Habilidad de la librería para el manejo de subrutinas de
interrupciones.*/

#include <avr/interrupt.h>

/* Definición y establecimiento de la frecuencia del oscilador
externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL
#endif

/* Habilidad de la librería para el manejo de retardos.*/

#include <avr/delay.h>

//*****
//* 2) DECLARACIÓN DE VARIABLES GLOBALES.
//*****

/* Unicamente se define una variable global que se utiliza en el
conteo de segundos para encendido o apagado de la pantalla del
```

```

LCD.*/

unsigned int segundos_lcd = 0;

//*****
/** 3) DECLARACIÓN DE FUNCIONES.
//*****

/* Se define unicamente una función que utilizada para configuración
del timer/counter 1.*/

void inic_timer_counter_1(void);

//*****
/** 4) FUNCIÓN DE CONFIGURACIÓN DEL TIMER/COUNTER 1.
//*****

void inic_timer_counter_1(void)
{
    /* Determinación del valor del registro OCR1A, que permita el
    conteo de 1 segundo cuando el valor de este registro sea igual
    al valor del contador TCNT1, caso en el cual se activa la
    interrupción de TIMER1_COMPA.*/

    #define TIMER_COUNTER_1_OCR1A_CALC(TOC1A_,FREQ_,N_)
    (((TOC1A_)*(FREQ_))/((N_)*2L)-1)*2L
    #define PERIODO_TOC1A      1// en segundos
    #define VALOR_PRESCALAMIENTO_N_A  64 // este valor se debe configurar en los
parametros del Timer/Counter1.

    OCR1A = TIMER_COUNTER_1_OCR1A_CALC( PERIODO_TOC1A, F_CPU,
VALOR_PRESCALAMIENTO_N_A);

    /* Determinación del valor del registro OCR1B, que permita el
    conteo de 1 segundo cuando el valor de este registro sea igual
    al valor del contador TCNT1, caso en el cual se activa la
    interrupción de TIMER1_COMPB.*/

    #define TIMER_COUNTER_1_OCR1B_CALC(TOC1B_,FREQ_,N_)
    (((TOC1B_)*(FREQ_))/((N_)*2L)-1)*2L
    #define PERIODO_TOC1B      1// en segundos//0.8
    #define VALOR_PRESCALAMIENTO_N_B  64 // este valor se debe configurar en los
parametros del Timer/Counter1.

    OCR1B = TIMER_COUNTER_1_OCR1B_CALC( PERIODO_TOC1B, F_CPU,
VALOR_PRESCALAMIENTO_N_B);

    /* Definir los parametros de configuración del Timer/Counter1:
    *
    * - Cambio de transición (modo toggle) en el pin OC1A cuando se
    * detecte que los valores entre el registro OCR1A y TCNT1
    * son iguales.
    * - Funcionamiento normal del pin correspondiente a OC1B.
    * - Reseteo del contador TCNT1 cuando el valor del registro
    * OCR1A y del TCNT1 son iguales.
    * - Valor de pre-escalamiento del reloj del Timer/Counter 1
    * en 64.
    *
    */

    TCCR1B = (0<<WGM13) | (1<<WGM12) | (0<<CS12) | (1<<CS11) | (1<<CS10);

```

```

        TCCR1A = (0<<COM1A1) | (1<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<WGM11) |
        (0<<WGM10);
    }

    /*******
    /*** 5) FUNCIÓN DE INTERRUPCIÓN DEL COMPARADOR A DEL TIMER/COUNTER 1.
    /*******

ISR(TIMER1_COMPA_vect)
{
    /* Desactivación de las interrupciones globales.*/

    cli();

    /* Detener el contador TCNT1. */

    TCCR1B &= ~(1<<CS12);
    TCCR1B &= ~(1<<CS11);
    TCCR1B &= ~(1<<CS10);

    /* Suma de segundos.*/

    segundos_lcd++;

    /* Receteo del contador TCNT1.*/

    TCNT1 = 0x00;

    /* Activar nuevamente el contador TCNT1 con un valor de
    preescalamiento de 64.*/

    TCCR1B &= ~(1<<CS12);
    TCCR1B |= (1<<CS11);
    TCCR1B |= (1<<CS10);

    /* Activación de las interrupciones globales.*/

    sei();
}

    /*******
    /*** 6) FUNCIÓN DE INTERRUPCIÓN DEL COMPARADOR B DEL TIMER/COUNTER 1.
    /*******

ISR(TIMER1_COMPB_vect)
{
    /* Desactivación de las interrupciones globales.*/

    cli();

    /* Detener el contador TCNT1. */

    TCCR1B &= ~(1<<CS12);
    TCCR1B &= ~(1<<CS11);
    TCCR1B &= ~(1<<CS10);

    /* Receteo del contador TCNT1.*/

    TCNT1 = 0x00;

    /* Activar nuevamente el contador TCNT1 con un valor de

```

```

        preescalamiento de 64.*/

TCCR1B &= ~(1<<CS12);
TCCR1B |= (1<<CS11);
TCCR1B |= (1<<CS10);

/* Activación de las interrupciones globales.*/

sei();

/* Habilitación de la conversión ADC de modo que una vez que
   la conversión se realice se procede a ejecutar la interrupción
   ADC. De esta forma la conversión ADC se realiza cuando se
   detecta una interrupción debido a la comparación del
   Timer/Counter1 con el registro OCR1B.*/

/* Activación de la conversión ADC.*/

ADCSRA |= (1 << ADEN);

/* Retardo de tiempo hasta que la conversión ADC concluya.*/

while ( ADCSRA & (1<<ADSC) )
{
    ;
}

#endif /* MUESTREO_H_ */

```

## A.5. PROGRAMA DE DETECCIÓN DE INTERRUPCIONES EXTERNAS PARA EL ENCENDIDO DEL LCD Y CONTEO DE BASCULACIONES DEL SENSOR DE PRECIPITACIÓN

```

/* ////////////////////////////////////////
 *
 * NOMBRE DEL PROYECTO: interrupcion_externa.h
 * FUNCIÓN: Librería que permite el manejo de un LCD de
 *          4x20 caracteres.
 * FECHA DE CREACIÓN: 30/11/2011
 * AUTOR: Miguel Angel Bonilla Vergara.
 *
 * ////////////////////////////////////////
 //*****
 //** 1) DEFINICIÓN DE LIBRERÍAS.
 //*****
 /* Definición de la librería uart.h a ser utilizado en el programa
    principal.*/

#ifndef INTERRUPCION_EXTERNA_H_
#define INTERRUPCION_EXTERNA_H_

/* Habilitación de la librería para el manejo de las entradas/salidas
   digitales.*/

```

```

#include <avr/io.h>

/* Habilitación de la librería para el manejo de variables tipo
   enteras.*/

#include <stdint.h>

/* Habilitación de la librería para el manejo de subrutinas de
   interrupciones.*/

#include <avr/interrupt.h>

/* Definición y establecimiento de la frecuencia del oscilador
   externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL
#endif

/* Habilitación de la librería para el manejo de retardos.*/

#include <avr/delay.h>

/*****
/* 2) DEFINICIÓN DE LIBRERIAS PREVIAMENTE ESTRUCTURADAS.
*****/

/* Se define la función para manejo y control del LCD.*/

#include "lcd.h"

/*****
/* 3) DECLARACIÓN DE VARIABLES GLOBALES.
*****/

/* Se define variables globales para el conteo de pulsos y la
   conversión de valores numéricos en caracteres ASCII.*/

char caracteres_pulsos_INT0_obtenidos[7];
unsigned int contador_pulsos = 0;

/*****
/* 4) DECLARACIÓN DE FUNCIONES.
*****/

/* Se define unicamente una función que utilizada para lectura de
   los pulsos obtenidos de las basculaciones del sensor de
   precipitación.*/

char * lectura_pulsos_reed_switch(unsigned int pulsos);

/*****
/* 5) FUNCIÓN DE INTERRUPCIÓN EXTERNA INT0.
*****/

ISR(INT0_vect)
{
    /* Desactivación de las interrupciones globales.*/

    cli();

```

```

/* Conteo de pulsos cuando se detecte un cambio de transición
en el pin correspondiente a la interrupción INT0.*/

contador_pulsos++;

_delay_ms(100);

/* Habilitación de las interrupciones globales.*/

sei();
}

//*****
/* 6) FUNCIÓN DE INTERRUPCIÓN EXTERNA INT1.
//*****

ISR(INT1_vect)
{
    /* Desactivación de las interrupciones globales.*/

    cli();
    _delay_ms(50);

    /* Declaración de variables locales para escritura de caracteres
en el LCD.*/

    char linea5[] = "ESTACION INIAQUITO";
    char linea6[] = "TEMPERATU: ";
    char linea7[] = "HUMEDAD : ";
    char linea8[] = "BATERIA A: ";
    char linea9[] = "oC";
    char linea10[] = "%";

    /* Reseteo del conteo de segundos y activación de la pantalla
del LCD.*/

    segundos_lcd = 0;
    PORTA |= (1 << PA4);

    /* Ejecución de comandos que son enviados al LCD para
activación del cursor, borrao de pantalla y ubicación
del cursor en posición inicial.*/

    comandos_LCD(0x0E);
    comandos_LCD(Borrar_Pantalla);
    comandos_LCD(Posicion_Inicial);

    /* Posicionamiento del cursor en la correspondiente fila Y
columna para la correspondiente escritura de los caracteres
previamente definidos.*/

    posicionar_cursor(1,2);
    escribir_cadena_caracteres_LCD(linea5);
    posicionar_cursor(2,1);
    escribir_cadena_caracteres_LCD(linea6);
    posicionar_cursor(3,1);
    escribir_cadena_caracteres_LCD(linea7);
    posicionar_cursor(4,1);
    escribir_cadena_caracteres_LCD(linea8);
    posicionar_cursor(2,19);

```

```

    escribir_cadena_caracteres_LCD(linea9);
    posicionar_cursor(3,20);
    escribir_cadena_caracteres_LCD(linea10);
    posicionar_cursor(4,20);
    escribir_cadena_caracteres_LCD(linea10);

    /* Habilitación de las interrupciones globales.*/

    _delay_ms(50);
    sei();
}

//*****
/* 7) FUNCIÓN DE LECTURA DE PULSOS.
//*****

char * lectura_pulsos_reed_switch(unsigned int pulsos)
{
    /* Declaración de variables locales.*/

    float contador_flotante = 0;

    /* Conversión en valores flotantes de los pulsos detectados.*/

    contador_flotante = (float)(pulsos);

    /* Se divide para 10 para indicar la cantidad de precipitación
       en mm por minuto.*/

    contador_flotante = contador_flotante/10;

    /* Conversión de la cantidad de precipitación en mm, a
       cadena de caracteres para su retorno al programa principal y
       presentación en el LCD.*/

    dtostrf(contador_flotante, 3, 1, caracteres_pulsos_INT0_obtenidos);
    return caracteres_pulsos_INT0_obtenidos;
}

#endif /* INTERRUPCION_EXTERNA_H_ */

```

## A.6. PROGRAMA DE MANEJO Y CONTROL DEL INTERFAZ USART DEL MICROCONTROLADOR, PARA TRANSMISIÓN Y RECEPCIÓN DE INFORMACIÓN DEL SENSOR DE TEMPERATURA, HUMEDAD Y NIVEL DE BATERÍA A

```

/* //////////////////////////////////////
 *
 * NOMBRE DEL PROYECTO: uart.h
 * FUNCIÓN: Librería que permite realizar la transmisión y recepción
 *          de datos asincrónica, mediante el uso del interfaz USART.
 * FECHA DE CREACIÓN: 23/11/2011
 * AUTOR: Miguel Angel Bonilla Vergara.
 *
 * //////////////////////////////////////

```

```

/*****
/* 1) DEFINICIÓN DE LIBRERÍAS.
*****/

/* Definición de la librería uart.h a ser utilizado en el programa
principal.*/

#ifndef UART_H_
#define UART_H_

/* Habilidad de la librería para el manejo de las entradas/salidas
digitales.*/

#include <avr/io.h>

/* Habilidad de la librería para operaciones con cadenas de
caracteres.*/

#include <string.h>

/* Habilidad de la librería para el manejo de subrutinas de
interrupciones.*/

#include <avr/interrupt.h>

/* Habilidad de la librería para el manejo de variables tipo
enteras.*/

#include <inttypes.h>

/* Definición y establecimiento de la frecuencia del oscilador
externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL
#endif

/* Habilidad de la librería para el manejo de retardos.*/

#include <avr/delay.h>

/*****
/* 2) DECLARACIÓN DE VARIABLES Y CONSTANTES GLOBALES.
*****/

/* Las variables y constantes globales a ser utilizadas tienen como
* tarea:
*
* - Dimensionar la longitud del buffer de recepción para datos de
* temperatura.
* - Dimensionar la longitud del buffer de recepción para datos de
* humedad.
* - Establecer el tiempo de transmisión.
* - Conteo de caracteres contenidos en los correspondientes
* buffers de transmisión y recepción.
* - Lectura de los caracteres de un determinado buffer.
* - Bandera de recepción que indica que información de
* temperatura o humedad se ha recibido.
*
*/

```

```

#define Longitud_Buffer_RX 10
#define Tiempo_Tx 10

volatile char Buffer_RX_TR[Longitud_Buffer_RX];
volatile char Buffer_RX_HR[Longitud_Buffer_RX];
unsigned int contador_buffer_Rx = 0;
char *cadena_caracteres_recibidos;
unsigned int bandera_Rx = 0;

/*****
/* 3) DECLARACIÓN DE FUNCIONES.
*****/

/* Las funciones que se definen a continuación presentan las
* siguientes tareas:
*
* - Configuración del interfaz USART.
* - Envío de datos de temperatura.
* - Envío de datos de humedad.
* - Envío del porcentaje de nivel de batería.
* - Recepción de datos.
*/

void inic_USART(void);
void enviar_datos_temperatura_al_buffer_TX(char* text);
void enviar_datos_humedad_al_buffer_TX(char* text);
void enviar_datos_nivel_bateria_al_buffer_TX(char* text);
char * recibir_datos_buffer_RX(char caracteres_TR_HR);

/*****
/* 4) FUNCIÓN DE CONFIGURACIÓN DEL USART.
*****/

void inic_USART(void)
{
    /* Determinación del valor del registro UBRR0, para una velocidad
    de transmisión de 9600bps y un oscilador de 4MHz.*/

    #define UART_UBRR_CALC(BAUD_,FREQ_) ((FREQ_)/((BAUD_)*16L)-1)
    #define UART_BAUD_RATE 9600

    UBRR0 = UART_UBRR_CALC(UART_BAUD_RATE, F_CPU)+1;

    /* Establecer el formato de trama que se utilizará en el
    interfaz USART. */

    /* La trama esta estructura de acuerdo a los siguientes
    * parámetros:
    *
    * - Modo Asíncrono USART.
    * - Modo de paridad des-habilitada.
    * - 1-bit de parada.
    * - 8-bits de datos.
    * - Desactivación de paridad.
    */

    UCSR0C = (0<<UMSEL01) | (0<<UMSEL00) | (0<<UPM01) | (0<<UPM00) | (0<<USBS0) |
    (0<<UCSZ02) | (1<<UCSZ01) | (1<<UCSZ00) | (1<<UCPOL0);

```

```

    UCSRA = 0B0000000;

    /* Habilitación de los pines de transmisión y recepción. */

    UCSRB |= (1<<TXEN0);
    UCSRB |= (1<<RXEN0);
}

//*****
/* 5) FUNCIÓN DE TRANSMISIÓN DE DATOS DEL SENSOR DE TEMPERATURA.
//*****

void enviar_datos_temperatura_al_buffer_TX(char* text)
{
    /* Declaración de variables locales.*/

    unsigned char i = 0;
    volatile unsigned int Tamano_Datos_TX = 0;

    /* Dimensionamiento del tamaño de datos que se requieren enviar.*/

    Tamano_Datos_TX = strlen(text);

    //para convertir de char a *char: (char*)&numero_flotante

    /* Se envía inicialmente el caracter T el cual indica que los
       caracteres a continuación del mismo son de temperatura.*/

    UDR0 = 'T';
    _delay_ms(Tiempo_Tx);

    /* Envío de caracter por caracter al correspondiente pin de
       transmisión de los valores de voltaje del sensor de
       temperatura obtenidos de la conversión ADC del canal AD0.*/

    for (i = 0; i < Tamano_Datos_TX; i++)
    {
        UDR0 = text[i];
        _delay_ms(Tiempo_Tx);
    }
}

//*****
/* 6) FUNCIÓN DE TRANSMISIÓN DE DATOS DEL SENSOR DE HUMEDAD.
//*****

void enviar_datos_humedad_al_buffer_TX(char* text)
{
    /* Declaración de variables locales. */

    unsigned char i = 0;
    volatile unsigned int Tamano_Datos_TX = 0;

    /* Dimensionamiento del tamaño de datos que se requieren enviar.*/

    Tamano_Datos_TX = strlen(text);

    //para convertir de char a *char: (char*)&numero_flotante

    /* Se envía inicialmente el caracter H el cual indica que los

```

```

    caracteres a continuación del mismo son de humedad.*/

UDR0 = 'H';
_delay_ms(Tiempo_Tx);

/* Envío de caracter por caracter al correspondiente pin de
transmisión de los valores de voltaje del sensor de
humedad obtenidos de la conversión ADC del canal AD2.*/

for (i = 0; i < Tamano_Datos_TX; i++)
{
    UDR0 = text[i];
    _delay_ms(Tiempo_Tx);
}
}

//*****
/* 7) FUNCIÓN DE TRANSMISIÓN DEL PORCENTAJE DEL NIVEL DE BATERÍA.
//*****

void enviar_datos_nivel_bateria_al_buffer_TX(char* text)
{
    /* Declaración de variables locales. */

    unsigned char i = 0;
    volatile unsigned int Tamano_Datos_TX = 0;

    /* Dimensionamiento del tamaño de datos que se requieren enviar.*/

    Tamano_Datos_TX = strlen(text);

    //para convertir de char a *char: (char*)&numero_flotante

    /* Se envía inicialmente el caracter A el cual indica que los
caracteres a continuación del mismo son del porcentaje de
nivel de Batería A.*/

    UDR0 = 'A';
    _delay_ms(Tiempo_Tx);

    /* Envío de caracter por caracter al correspondiente pin de
transmisión de los valores de porcentaje del nivel de
batería A correspondiente al módulo de temperatura y
humedad relativa.*/

    for (i = 0; i < Tamano_Datos_TX; i++)
    {
        UDR0 = text[i];
        _delay_ms(Tiempo_Tx);
    }
}

//*****
/* 8) FUNCIÓN DE INTERRUPCIÓN DE RECEPCIÓN DEL USART0.
//*****

ISR(USART0_RX_vect)
{
    /* Desactivación de las interrupciones globales.*/

    cli();
}

```

```

/* Declaración de la variable local en la cual se almacena el
   caracteres que es recibido.*/

unsigned char datos_RX;

/* Recepción de caracter por caracter mediante el registro UDR0.*/

datos_RX = UDR0;

/* RECEPCIÓN DE LOS CARACTERES DE TEMPERATURA.*/

/* Una vez que la bandera de Rx se haya activado, se verifica
   que el dato leído no sea el caracter F, en cuyo caso la trama
   de datos ha terminado y por ende se debe recetear la bandera
   Rx, así como el contador del buffer Rx. Caso contrario, se
   procede a almacenar los datos recibidos en el buffer Rx.*/

if (bandera_Rx == 1)
{
    if (datos_RX == 'F')
    {
        /* Receteo de la bandera y buffer de Rx una vez que la
           trama de datos recibidos han sido leídos.*/

        bandera_Rx = 0;
        contador_buffer_Rx = 0;
        datos_RX = 0;
    }

    else
    {
        /* Se almacena los datos de temperatura recibidos en un
           buffer Rx.*/

        contador_buffer_Rx++;
        Buffer_RX_TR[contador_buffer_Rx - 1] = datos_RX;
    }
}

/* Se utiliza una bandera de Rx para indicar los datos que queremos
   mostrar en el LCD. Adicional se verifica si el dato leído es T
   al ultimo de la secuencia para almacenar los datos numericos en
   la siguiente iteracion, es decir que no se permite que el primer
   parametro T se muestre en el display.*/

/* Establecer un valor de 1 a la bandera Rx, para indicar que los
   datos recibidos son de temperatura.*/

if (datos_RX == 'T') bandera_Rx = 1;

/* Vaciar el buffer RX. */

while (UCSR0A & (1<<RXC0));
//dummy = UDR0;

/* RECEPCIÓN DE LOS CARACTERES DE HUMEDAD.*/

/* Una vez que la bandera de Rx se haya activado, se verifica
   que el dato leído no sea el caracter F, en cuyo caso la trama
   de datos ha terminado y por ende se debe recetear la bandera

```

```

    Rx, así como el contador del buffer Rx. Caso contrario, se
    procede a almacenar los datos recibidos en el buffer Rx.*/

if (bandera_Rx == 2)
{
    if (datos_RX == 'F')
    {
        /* Receteo de la bandera y buffer de Rx una vez que la
        trama de datos recibidos han sido leídos.*/

        bandera_Rx = 0;
        contador_buffer_Rx = 0;
        datos_RX = 0;
    }

    else
    {
        /* Se almacena los datos de humedad recibidos en un
        buffer Rx.*/

        contador_buffer_Rx++;
        Buffer_RX_HR[contador_buffer_Rx - 1] = datos_RX;
    }
}

/* Se utiliza una bandera de Rx para indicar los datos que queremos
mostrar en el LCD. Adicional se verifica si el dato leído es T
al ultimo de la secuencia para almacen los datos numericos en
la siguiente iteracion, es decir que no se permite que el primer
parametro T se muestre en el display.*/

/* Establecer un valor de 2 en la bandera Rx, para indicar que los
datos recibidos son de humedad.*/

if (datos_RX == 'H') bandera_Rx = 2;

/* Vaciar el buffer RX. */
while (UCSR0A & (1<<RXC0));
//dummy = UDR0;

/* Activación de las interrupciones globales.*/

sei();
}

/*****
/* 9) FUNCIÓN DE RECEPCIÓN DE DATOS RX.
*****/

char * recibir_datos_buffer_RX(char caracteres_TR_HR)
{
    /* Almacenamiento de la cadena de caracteres correspondientes a
    los valores de temperatura procesados en una variable
    global.*/

    if (caracteres_TR_HR == 'T') cadena_caracteres_recibidos =
(char*)&Buffer_RX_TR;

    /* Almacenamiento de la cadena de caracteres correspondientes a
    los valores de humedad procesados en una variable global.*/

```

```

        if (caracteres_TR_HR == 'H') cadena_caracteres_recibidos =
            (char*)&Buffer_RX_HR;
        //cadena_caracteres_recibidos = (char*)&Buffer_RX_TR;

        /* Retorno de la cadena de caracteres de los valores de
           temperatura o humedad procesados, al programa principal.*/

        return cadena_caracteres_recibidos;
    }

#endif /* UART_H_ */

```

## B. PROGRAMAS DE PROCESAMIENTO DE INFORMACIÓN DEL MÓDULO DE PRECIPITACIÓN

### B.1. PROGRAMA PRINCIPAL DEL MÓDULO DE PRECIPITACIÓN

```

/* ////////////////////////////////////////
 *
 * NOMBRE DEL PROYECTO: Adqusicion_Precip.cpp
 * FUNCIÓN: Programa principal que permite adquirir y procesar los
 *         valores de basculaciones del sensor de precipitación,
 *         para su posterior envío al correspondiente dispositivo
 *         ZigBee y presentación en el LCD.
 * FECHA DE CREACIÓN: 12/06/2013
 * AUTOR: Miguel Angel Bonilla Vergara.
 *
 * ////////////////////////////////////////
//*****
//** 1) DEFINICIÓN DE LIBRERÍAS.
//*****

/* Habilitación de la librería para el manejo de las entradas/salidas
   digitales.*/

#include <avr/io.h>

/* Habilitación de la librería para la conversión de cadenas de
   caracteres y valores numéricos.*/

#include <stdlib.h>

/* Habilitación de la librería para operaciones matemáticas básicas.*/

#include <math.h>

/* Habilitación de la librería para operaciones con cadenas de
   caracteres.*/

#include <string.h>

/* Habilitación de la librería para el manejo de variables tipo

```

```

    enteras.*/

#include <stdint.h>

/* Habilidad de la librería para el manejo de subrutinas de
   interrupciones.*/

#include <avr/interrupt.h>

/* Definición y establecimiento de la frecuencia del oscilador
   externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL
#endif

/* Habilidad de la librería para el manejo de retardos.*/

#include <avr/delay.h>
#include <util/delay.h>

/*****
/* 2) DEFINICIÓN DE LIBRERIAS PREVIAMENTE ESTRUCTURADAS.
*****/

/* Habilidad de las librerías para el manejo de los siguientes
   funciones:
   *
   * - Conversión ADC simple.
   * - Manejo del LCD.
   * - Muestreo de la conversión ADC.
   * - Manejo de interrupciones externas.
   * - Manejo del interfaz UART.
   *
   */

#include "adc.h"
#include "lcd.h"
#include "muestreo.h"
#include "uart.h"
#include "interrupcion_externa.h"

/*****
/* 3) DECLARACIÓN DE VARIABLES GLOBALES.
*****/

/* Las variables globales a ser utilizadas tienen como tarea:
   *
   * - Obtención de la cadena de caracteres que indica el porcentaje
   *   del nivel de batería de funcionamiento del circuito.
   *
   */

char caracteres_canal_ADC_obtenidos[7];

/*****
/* 4) DECLARACIÓN DE FUNCIONES.
*****/

/* Las funciones que se definen a continuación presentan las
   * siguientes tareas:

```

```

*
* - Inicialización y configuración de los pórtricos de entrada
* y salida digitales de proposito general.
* - Inicialización de las interrupciones de los subsistemas
* que se utilizarán durante la ejecución del programa.
* - Lectura del porcentaje de batería para funcionamiento del
* circuito.
*
*/

void inic_porticos(void);
void inic_interrupciones(void);
char * lectura_nivel_bateria(void);

//*****
/* 5) PROGRAMA PRINCIPAL.
//*****

int main(void)
{
    /* Llamado y ejecución de funciones previamente declaradas y
    correspondientes a:
    *
    * - Inicialización de pórtricos de entradas/salidas digitales.
    * - Inicialización del manejo del LCD.
    * - Inicialización de la conversión ADC simple.
    * - Inicialización del temporizador contador 1.
    * - Inicialización del interfaz UART.
    * - Inicialización de las interrupciones externas.
    *
    */

    inic_porticos();
    inic_LCD();
    inic_ADC_simple();
    inic_timer_counter_1();
    inic_USART();
    inic_interrupciones();

    /* Encendido de la pantalla del LCD.*/

    PORTA |= (1 << PA4);

    /* Declaración de variables locales las cuales son utilizadas
    * para:
    *
    * - Presentación inicial a ser mostrada en el LCD.
    * - Lectura de caracteres de voltaje a ser enviados.
    * - Lectura de los caracteres recibidos.
    *
    */

    char linea1[] = "EPN-INAMHI ";
    char linea3[] = "MOD. PRECIPITACION";
    char linea4[] = "MIGUEL BONILLA";
    char *caracteres_voltajes_obtenidos;
    char *caracteres_pulsos_obtenidos;

    /* Ejecución de comandos que son enviados al LCD para
    activación del cursor, borrar de pantalla y ubicación
    del cursor en posición inicial.*/

```

```

comandos_LCD(Borrar_Pantalla);
comandos_LCD(Posicion_Inicial);

/* Posicionamiento del cursor en la correspondiente fila Y
   columna para la correspondiente escritura de los caracteres
   previamente definidos.*/

posicionar_cursor(1,6);
escribir_cadena_caracteres_LCD(linea1);
posicionar_cursor(3,2);
escribir_cadena_caracteres_LCD(linea3);
posicionar_cursor(4,4);
escribir_cadena_caracteres_LCD(linea4);

/* Inicialización del contador de pulsos.*/

contador_pulsos = 0; //inicializar el contador

/* Ejecución de un retardo correspondiente a 9 segundos.*/

_delay_ms(9000);

/* Desactivación del LCD y apagado de la pantalla.*/

comandos_LCD(0x0A);
PORTA &= ~(1 << PA4);

while(1)
{
    /* Lectura del número de pulsos detectados del sensor de
       precipitación, para su correspondiente presentación en
       el LCD y posterior transmisión mediante el uso del
       interfaz USART.*/

    caracteres_pulsos_obtenidos =
    lectura_pulsos_reed_switch(contador_pulsos);
    posicionar_cursor(3,12);
    escribir_cadena_caracteres_LCD(caracteres_pulsos_obtenidos);
    enviar_datos_precipitacion_al_buffer_TX(caracteres_pulsos_obtenidos);

    /* Lectura del porcentaje del nivel de batería para
       su correspondiente transmisión mediante el uso del
       interfaz USART y posterior presentación en el LCD.*/

    caracteres_voltajes_obtenidos = lectura_nivel_bateria();
    enviar_datos_nivel_bateria_al_buffer_TX(caracteres_voltajes_obtenidos);
    posicionar_cursor(4,12);
    escribir_cadena_caracteres_LCD(caracteres_voltajes_obtenidos);

    /* Verificación de que el conteo de segundos sea de
       1 minuto, luego del cual se procede a resetear el
       contador de pulsos y el conteo de segundos. De esta
       forma se obtiene una cantidad de precipitación en
       milímetros (mm) por minuto.*/

    if (segundos == 80)
    {
        contador_pulsos = 0;
        segundos = 0;
    }
}

```

```

        /* Verificación del conteo de segundos correspondiente a
        5 minutos, luego de los cuales se procede a desactivar
        el LCD y apagar la pantalla.*/

        if (segundos_lcd == 300)
        {
            comandos_LCD(0x0A);
            PORTA &= ~(1 << PA4);
        }
    }
}

//*****
/* 6) FUNCIÓN DE INICIALIZACIÓN DE PÓRTICOS.
//*****

void inic_porticos(void)
{
    /* Habilitación de la salida de datos del LCD.*/

    DDRC = 0xFF;
    PORTC = 0x00;

    /* Habilitación de la salida de datos RS, RW y E del LCD.*/

    DDRA |= (1<<PA5) | (1<<PA6) | (1<<PA7);
    PORTA &= ~(1<<PA5) | (1<<PA6) | (1<<PA7));

    /* Habilitación del canal de conversión AD1 como entrada
    digital.*/

    DDRA &= ~(1<<PA1);
    PORTA &= ~(1<<PA1);

    /* Habilitación del pin OC1A como salida digital.*/

    DDRD |= (1<<PD5);
    PORTD &= ~(1<<PD5);

    /* Habilitación del pin OC0A como salida digital.*/

    DDRB |= (1<<PB3);
    PORTB &= ~(1<<PB3);

    /* Habilitación de la interrupción externa INT0 como
    entrada digital de datos.*/

    DDRD &= ~(1<<PD2);
    PORTD &= ~(1<<PD2);

    /* Habilitación de la interrupción externa INT1 como
    entrada digital de datos.*/

    DDRD &= ~(1<<PD3);
    PORTD &= ~(1<<PD3);

    /* Habilitación del pin para encendido o apagado del LCD
    como salida digital de datos.*/

    DDRA |= (1<<PA4);

```

```

PORTA &= ~(1<<PA4);

/* Habilitación de la recepción USART0 como entrada
   digital de datos.*/

DDRD &= ~(1<<PD0);
PORTD &= ~(1<<PD0);

/* Habilitación de la transmisión USART0 como salida
   digital de datos.*/

DDRD |= (1<<PD1);
PORTD &= ~(1<<PD1);

/* Activacion Pull-ups.*/

DDRD &= ~((1<<PD4) | (1<<PD6) | (1<<PD7));
PORTD |= (1<<PD4) | (1<<PD6) | (1<<PD7);

DDRA &= ~((1<<PA0) | (1<<PA2) | (1<<PA3));
PORTA |= (1<<PA0) | (1<<PA2) | (1<<PA3);

DDRB &= ~((1<<PB0) | (1<<PB1) | (1<<PB2) | (1<<PB4) | (1<<PB5) | (1<<PB6) |
          (1<<PB7));
PORTB |= (1<<PB0) | (1<<PB1) | (1<<PB2) | (1<<PB4) | (1<<PB5) | (1<<PB6) |
          (1<<PB7);
}

//*****
/* 7) FUNCIÓN DE INICIALIZACIÓN DE INTERRUPCIONES.
//*****

void inic_interrupciones(void)
{
    /* Habilitación de las interrupciones globales.*/

    sei();

    /* Habilitación de la interrupción de conversión ADC simple.*/

    ADCSRA |= (1<<ADIE);

    /* Habilitación de la interrupción cuando se presenta la
       detección de comparación en OCR1B.*/

    TIMSK1 |= (1<<OCIE1B);

    /* Habilitación de la interrupción cuando se presenta la
       detección de comparación en OCR1A.*/

    TIMSK1 |= (1<<OCIE1A);

    /* Habilitación de la interrupción de recepción completa del
       interfaz USART0.*/

    UCSR0B |= (1<<RXCIE0);

    /* Habilitación de la interrupción externa INT0, la cual se
       activa con la detección de una transición positiva.*/

    EIMSK |= (1<<INT0);
}

```

```

EICRA |= (1<<ISC00) | (1<<ISC01);

/* Habilitación de la interrupción externa INT1, la cual se
   activa con la detección de una transición positiva.*/

EIMSK |= (1<<INT1);
EICRA |= (1<<ISC10) | (1<<ISC11);
}

//*****
/* 8) FUNCIÓN DE LECTURA DEL NIVEL DE VOLTAJE DE LA BATERÍA B.
//*****

char * lectura_nivel_bateria(void)
{
    /* Declaración de variables locales.*/

    float Nivel_Bateria = 0;
    float voltaje_canal_ADC = 0;

    /* Lectura del voltaje del canal AD1.*/

    voltaje_canal_ADC = lectura_canal_ADC(1);

    /* Comprobación de que el nivel de voltaje sea mayor a 1.3,
       el cual representa el voltaje mínimo de operación.*/

    if (voltaje_canal_ADC > 1.3)
    {
        /* Determinar el voltaje de entrada al conversor ADC,
           tomando en cuenta que el rango de operación se
           encuentra entre 2.5V y 1.3V.*/

        voltaje_canal_ADC = voltaje_canal_ADC - 1.3;
    }
    else
    {
        /* En este caso el voltaje de operación es menor a
           al valor mínimo, por lo cual el porcentaje debe
           ser igual a cero.*/

        voltaje_canal_ADC = 0;
    }

    /* Determinación del nivel de batería en porcentaje.*/

    Nivel_Bateria = (voltaje_canal_ADC * 100)/1.26;

    /* Conversión del nivel de batería en una cadena de caracteres.*/

    dtostrf( Nivel_Bateria, 3, 0, caracteres_canal_ADC_obtenidos );

    /* Retorno de la cadena de caracteres del nivel de batería al
       programa principal.*/

    return caracteres_canal_ADC_obtenidos;
}

```

## B.2. PROGRAMA DE CONVERSIÓN ANALÓGICO A DIGITAL, CORRESPONDIENTE A LA LECTURA DE UN SOLO CANAL AD DEL MICROPROCESADOR

```
/* ////////////////////////////////////////
 *
 * NOMBRE DEL PROYECTO: adc.h
 * FUNCIÓN: Librería que permite realizar la conversión ADC simple
 *         del canal AD1.
 * FECHA DE CREACIÓN: 24/11/2013
 * AUTOR: Miguel Angel Bonilla Vergara.
 *
 * ////////////////////////////////////////

//*****
//* 1) DEFINICIÓN DE LIBRERÍAS.
//*****

/* Definición de la librería adc.h a ser utiliza en el programa
   principal.*/

#ifndef ADC_H_
#define ADC_H_

/* Habilidad de la librería para el manejo de las entradas/salidas
   digitales.*/

#include <avr/io.h>

/* Habilidad de la librería para el manejo de variables tipo
   enteras.*/

#include <stdint.h>

/* Habilidad de la librería para la conversión de cadenas de
   caracteres y valores numéricos.*/

#include <stdlib.h>

/* Habilidad de la librería para operaciones con cadenas de
   caracteres.*/

#include <string.h>

/* Habilidad de la librería para el manejo de subrutinas de
   interrupciones.*/

#include <avr/interrupt.h>

/* Definición y establecimiento de la frecuencia del oscilador
   externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL /* cristal de 4MHz */
#endif

/* Habilidad de la librería para el manejo de retardos.*/
```

```

#include <avr/delay.h>

/*****
/* 2) DECLARACIÓN DE VARIABLES GLOBALES.
*****/

/* Las variables globales a ser utilizadas tienen como tarea:
*
* - Establecer una bandera ADC que indica la conversión del
* siguiente canal ADC.
* - Obtención del valor flotante del canal AD0.
* - Obtención del valor flotante del canal AD1.
*
*/

uint8_t bandera_canal_ADC = 0;
float voltaje_AD0 = 0;
float voltaje_AD1 = 0 ;

/*****
/* 3) DECLARACIÓN DE FUNCIONES.
*****/

/* Las funciones que se definen a continuación presentan las
* siguientes tareas:
*
* - Configuración de la conversión ADC simple.
* - Lectura del valor del canal ADC.
*
*/

void inic_ADC_simple(void);
float lectura_canal_ADC(unsigned char canal_ADC);

/*****
/* 4) FUNCIÓN DE CONFIGURACIÓN DEL SUBSISTEMA ADC DEL AVR.
*****/

void inic_ADC_simple (void)
{
    /* Configuración de la Conversión ADC Simple.*/

    /* La configuración de los registros bit por bit del ADC,
    * permite que se realice las siguientes funciones:
    *
    * - Habilitación del ADC.
    * - Habilitación del modo de auto-conversión con una fuente
    * FREE-RUNNING.
    * - Establecimiento de un valor de preescalamiento de 8 para
    * obtener una frecuencia de reloj para operación del ADC (clkADC)
    * entre 50KHz y 200kHz.
    * - Establecimiento de un voltaje de referencia interno de 2.56V.
    * - Habilitación del ajuste de resultado en la derecha.
    * - Habilitación del canal 0 (AD0) para el ingreso de una señal
    * analógica.
    * - Des-habilitación de los canales ADC que no se van a utilizar.
    *
    */

    ADCSRA = (1<<ADEN) | (1<<ADSC) | (0<<ADIFSC) | (0<<ADIFR) | (1<<ADIF0) |

```

```

        (1<<ADPS0);
ADCSRB = (1<<ADTS2) | (0<<ADTS1) | (1<<ADTS0);
ADMUX  = (1<<REFS1) | (1<<REFS0) | (0<<ADLAR);

/* Habilitación del canal AD0 que inicialmente se va a leer.*/

ADMUX  &= ~(1<<MUX4);
ADMUX  &= ~(1<<MUX3);
ADMUX  &= ~(1<<MUX2);
ADMUX  &= ~(1<<MUX1);
ADMUX  &= ~(1<<MUX0);

/* Desactivación de las entradas digitales de los canales en los
   que se realice la conversión ADC.*/

DIDR0  = (0<<ADC7D) | (0<<ADC6D) | (0<<ADC5D) | (0<<ADC4D) | (0<<ADC3D) |
        (0<<ADC2D) | (1<<ADC1D) | (0<<ADC0D);
}

/*****
/* 5) FUNCIÓN DE LECTURA DE LOS CANALES ADC.
*****/

float lectura_canal_ADC(unsigned char canal_ADC)
{
    /* Declaración de variables locales.*/

    float voltaje_ADC = 0;

    /* Lectura del canal AD0.*/

    if (canal_ADC == 0)
    {
        voltaje_ADC = voltaje_AD0;
    }

    /* Lectura del canal AD1.*/

    if (canal_ADC == 1)
    {
        voltaje_ADC = voltaje_AD1;
    }

    /* Retorno del voltaje del canal ADC leído.*/

    return voltaje_ADC;
}

/*****
/* 6) FUNCIÓN DE INTERRUPCIÓN DE CONVERSIÓN ADC SIMPLE.
*****/

ISR(ADC_vect)
{
    /* Desactivación del subsistema ADC del AVR para tomar los
       valores de los registros ADC.*/

    ADCSRA &= ~(1 << ADEN);

    /* Declaración de variables locales para lectura de los
       registros correspondientes a los bits menos y más

```

```

    significativos, conversión ADC entero y flotante y
    el valor de voltaje de entrada flotante.*/

unsigned int registro_ADCL = 0;
unsigned int registro_ADCH = 0;
unsigned int conversion_ADC_entero = 0;
float conversion_ADC_flotante = 0;
float vin_flotante = 0;

/* Verificación de la bandera que indica que canal debe
ser leído.*/

if (bandera_canal_ADC == 1)
{
    /* Habilitación del siguiente canal AD1 a ser leído.*/

    ADMUX  &= ~(1<<MUX4);
    ADMUX  &= ~(1<<MUX3);
    ADMUX  &= ~(1<<MUX2);
    ADMUX  &= ~(1<<MUX1);
    ADMUX  |=  (1<<MUX0);

    /* Inicialmente se procede con la lectura de los 8-bits
    del registro ADCL.*/

    registro_ADCL = ADCL;

    /* A continuación se procede con la lectura de los 8-bits
    del registro ADCH.*/

    registro_ADCH = ((unsigned int)(ADCH << 8));

    /* Concatenación de la lectura de ambos registros en una sola
    variable, y conversión de esta variable en un solo valor
    flotante.*/

    conversion_ADC_entero = registro_ADCL | registro_ADCH;
    conversion_ADC_flotante = (float)(conversion_ADC_entero);

    /* Ecuación de conversión de un voltaje de entrada en el
    canal AD1, y su correspondiente almacenamiento en una
    variable local. */

    vin_flotante = (conversion_ADC_flotante * 2.56)/1024;
    voltaje_AD1 = vin_flotante;
}

/* La bandera ADC toma unicamente el valor de 1 ya que solo
se requiere leer el canal AD1.*/

bandera_canal_ADC = 1;
}

#endif /* ADC_H_ */

```

### B.3. PROGRAMA DE MANEJO Y CONTROL DEL INTERFAZ UART DEL MICROCONTROLADOR, PARA TRANSMISIÓN DE INFORMACIÓN DEL SENSOR DE PRECIPITACIÓN Y NIVEL DE BATERÍA B

```
/* ////////////////////////////////////////
 *
 * NOMBRE DEL PROYECTO: uart.h
 * FUNCIÓN: Librería que permite realizar la transmisión y recepción
 *         de datos asincrónica, mediante el uso del interfaz USART.
 * FECHA DE CREACIÓN: 23/11/2011
 * AUTOR: Miguel Angel Bonilla Vergara.
 *
 * ////////////////////////////////////////

//*****/
/* 1) DEFINICIÓN DE LIBRERÍAS.
//*****/

/* Definición de la librería uart.h a ser utiliza en el programa
principal.*/

#ifndef UART_H_
#define UART_H_

/* Habilitación de la librería para el manejo de las entradas/salidas
digitales.*/

#include <avr/io.h>

/* Habilitación de la librería para operaciones con cadenas de
caracteres.*/

#include <string.h>

/* Habilitación de la librería para el manejo de subrutinas de
interrupciones.*/

#include <avr/interrupt.h>

/* Habilitación de la librería para el manejo de variables tipo
enteras.*/

#include <inttypes.h>

/* Definición y establecimiento de la frecuencia del oscilador
externo en 4MHz.*/

#ifndef F_CPU
#define F_CPU 4000000UL
#endif

/* Habilitación de la librería para el manejo de retardos.*/

#include <avr/delay.h>
```

```

/*****
/** 2) DECLARACIÓN DE VARIABLES Y CONSTANTES GLOBALES.
*****/

/* La única constante que se define es la del tiempo de transmisión.*/

#define Tiempo_Tx 10

/*****
/** 3) DECLARACIÓN DE FUNCIONES.
*****/

/* Las funciones que se definen a continuación presentan las
 * siguientes tareas:
 *
 * - Configuración del interfaz USART.
 * - Envío de datos de temperatura.
 * - Envío de datos de humedad.
 * - Envío del porcentaje de nivel de batería.
 * - Recepción de datos.
 *
 */

void inic_USART(void);
void enviar_datos_precipitacion_al_buffer_TX(char* text);
void enviar_datos_nivel_bateria_al_buffer_TX(char* text);

/*****
/** 4) FUNCIÓN DE CONFIGURACIÓN DEL USART.
*****/

void inic_USART(void)
{
    /* Determinación del valor del registro UBRR0, para una velocidad
     * de transmisión de 9600bps y un oscilador de 4MHz.*/

    #define UART_UBRR_CALC(BAUD_,FREQ_) ((FREQ_)/((BAUD_)*16L)-1)
    #define UART_BAUD_RATE 9600

    UBRR0 = UART_UBRR_CALC(UART_BAUD_RATE, F_CPU)+1;

    /* Establecer el formato de trama que se utilizará en el
     * interfaz USART. */

    /* La trama esta estructura de acuerdo a los siguientes
     * parámetros:
     *
     * - Modo Asíncrono USART.
     * - Modo de paridad des-habilitada.
     * - 1-bit de parada.
     * - 8-bits de datos.
     * - Desactivación de paridad.
     *
     */

    UCSR0C = (0<<UMSEL01) | (0<<UMSEL00) | (0<<UPM01) | (0<<UPM00) | (0<<USBS0) |
    (0<<UCSZ02) | (1<<UCSZ01) | (1<<UCSZ00) | (1<<UCPOL0);
    UCSR0A = 0B00000000;

    /* Habilitación del pin de transmisión. */

```

```

        UCSR0B |= (1<<TXEN0);
    }

    /*******
    /** 5) FUNCIÓN DE TRANSMISIÓN DE DATOS DEL SENSOR DE PRECIPITACIÓN.
    /*******

void enviar_datos_precipitacion_al_buffer_TX(char* text)
{
    /* Declaración de variables locales.*/

    unsigned char i = 0;
    volatile unsigned int Tamano_Datos_TX = 0;

    /* Dimensionamiento del tamaño de datos que se requieren enviar.*/

    Tamano_Datos_TX = strlen(text);

    //para convertir de char a *char: (char*)&numero_flotante

    /* Se envía inicialmente el caracter P el cual indica que los
       caracteres a continuación del mismo son de precipitación.*/

    UDR0 = 'P';
    _delay_ms(Tiempo_Tx);

    /* Envío de caracter por caracter al correspondiente pin de
       transmisión de los valores de voltaje del sensor de
       temperatura obtenidos de la conversión ADC del canal AD0.*/

    for (i = 0; i < Tamano_Datos_TX; i++)
    {
        UDR0 = text[i];
        _delay_ms(Tiempo_Tx);
    }
}

    /*******
    /** 6) FUNCIÓN DE TRANSMISIÓN DEL PORCENTAJE DEL NIVEL DE BATERÍA.
    /*******

void enviar_datos_nivel_bateria_al_buffer_TX(char* text)
{
    /* Declaración de variables locales. */

    unsigned char i = 0;
    volatile unsigned int Tamano_Datos_TX = 0;

    /* Dimensionamiento del tamaño de datos que se requieren enviar.*/

    Tamano_Datos_TX = strlen(text);

    //para convertir de char a *char: (char*)&numero_flotante

    /* Se envía inicialmente el caracter A el cual indica que los
       caracteres a continuación del mismo son del porcentaje de
       nivel de Batería A.*/

    UDR0 = 'B';
    _delay_ms(Tiempo_Tx);
}

```

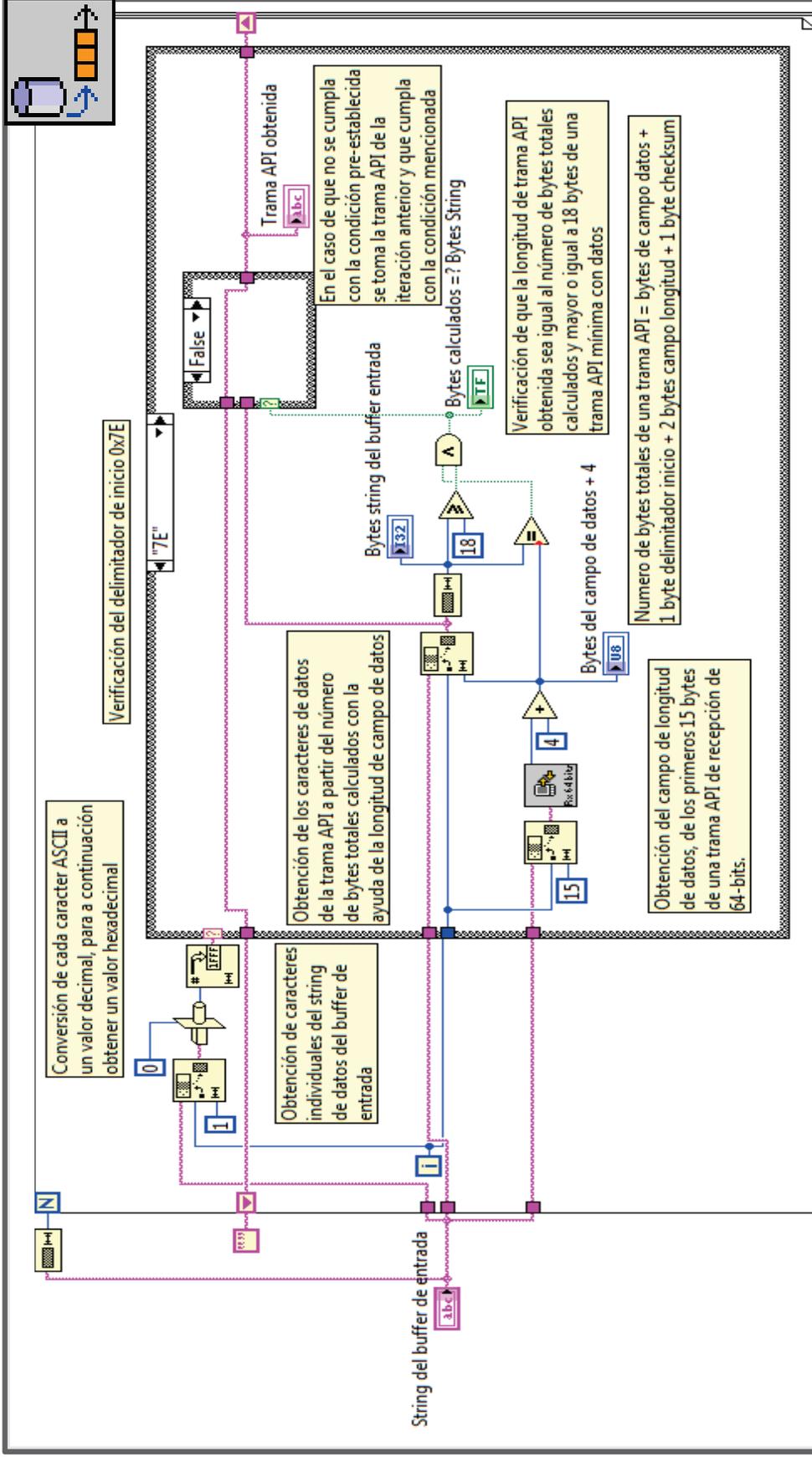
```
/* Envío de caracter por caracter al correspondiente pin de
transmisión de los valores de porcentaje del nivel de
batería A correspondiente al módulo de temperatura y
humedad relativa.*/

for (i = 0; i < Tamano_Datos_TX; i++)
{
    UDR0 = text[i];
    _delay_ms(Tiempo_Tx);
}

#endif /* UART_H_ */
```

## **C. DIAGRAMAS DE BLOQUES DEL DISEÑO DEL MÓDULO DE ADQUISICIÓN DE DATOS METEOROLÓGICOS**





**TÍTULO:** Trama API Obtenida del Buffer de Recepción

**MÓDULO:** Transmisión y Recepción Serial API

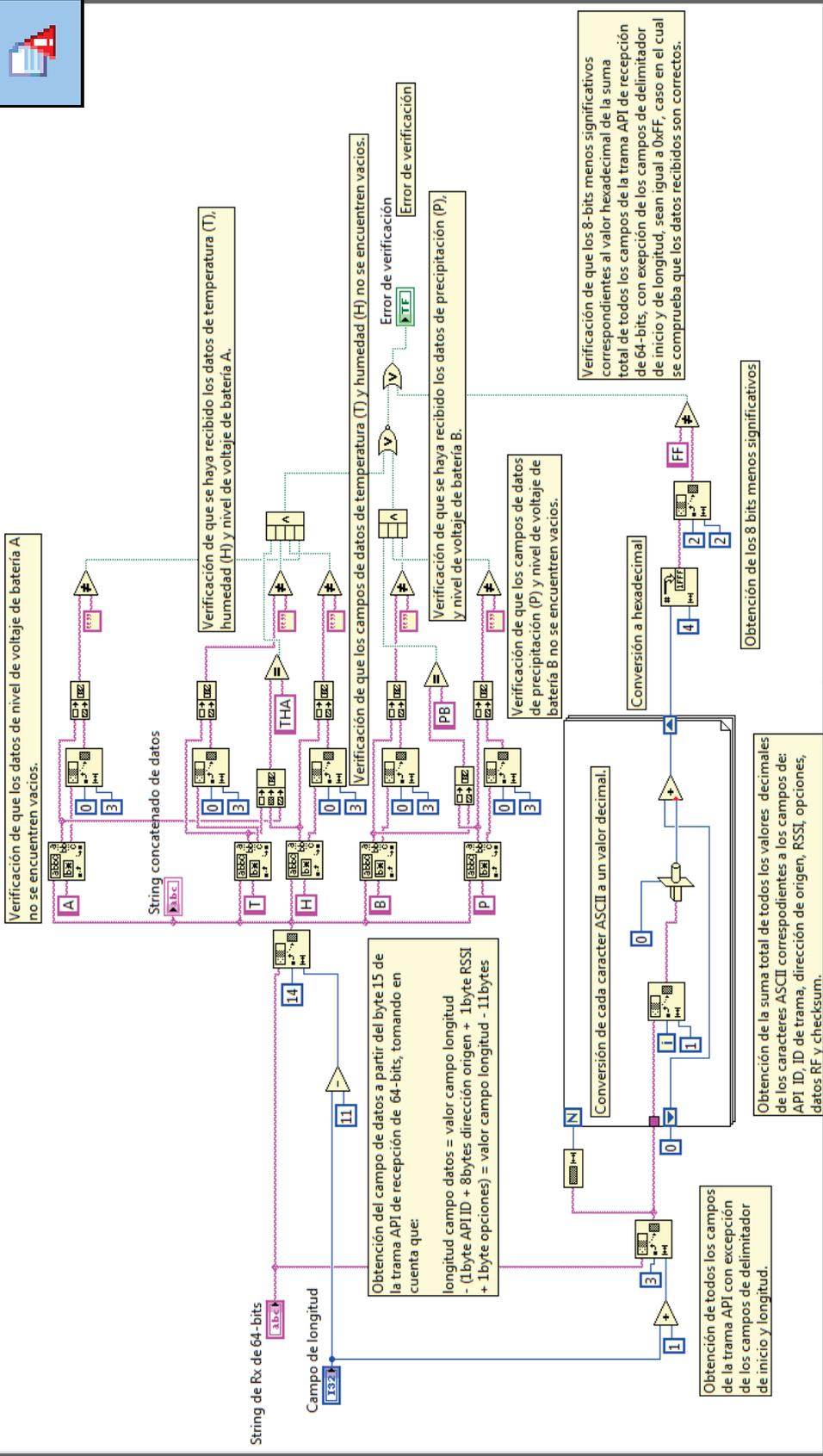
**ANEXO:** C1.1

**ESQUEMA No:** 2 de 6

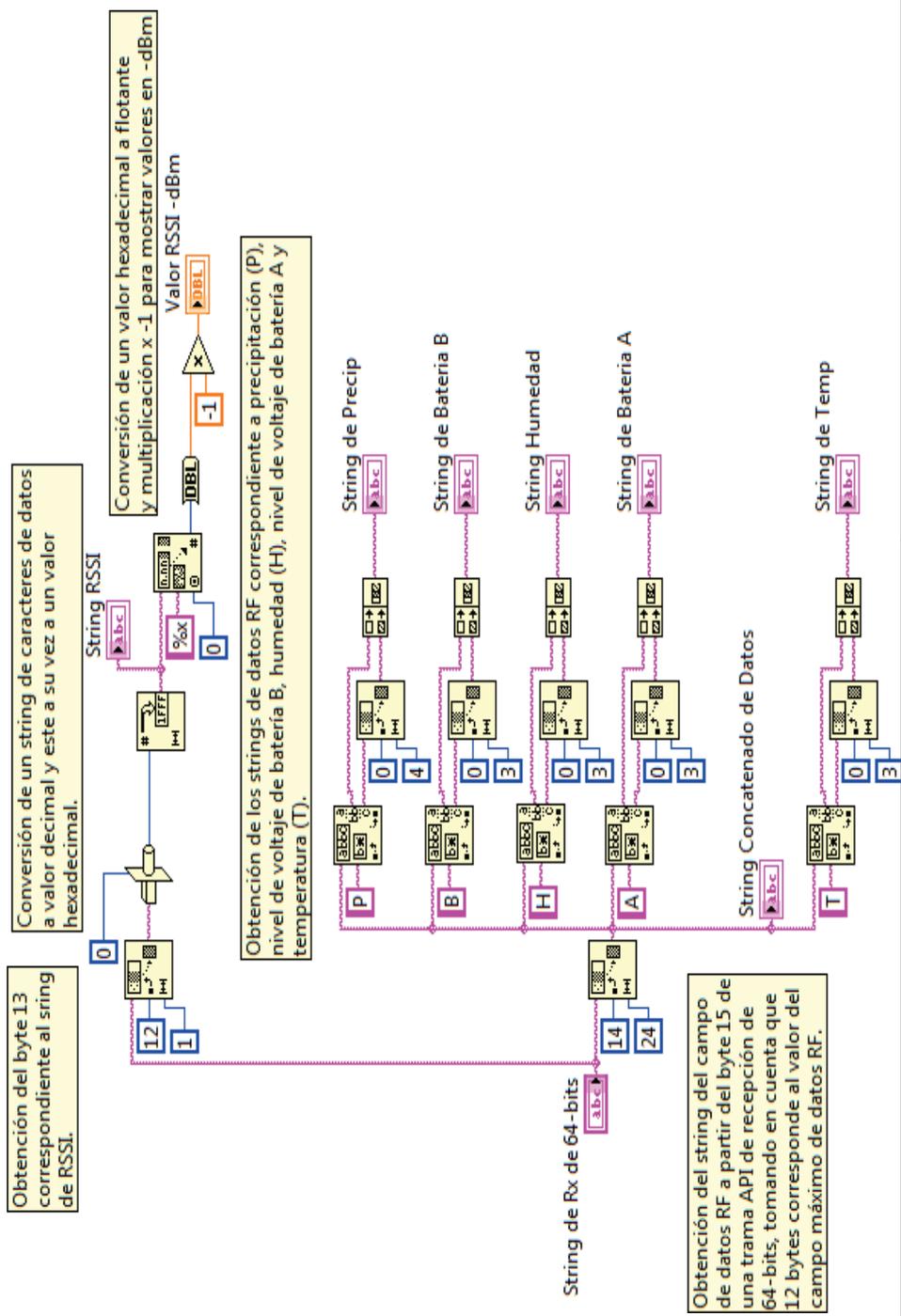
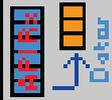
**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi





<b>TÍTULO: Verificación de Errores de la Trama API de Recepción</b>		<b>MÓDULO: Transmisión y Recepción Serial API</b>	
<b>ANEXO: C1.3</b>	<b>ESQUEMA No: 4 de 6</b>	<b>NOMBRE: Miguel Bonilla</b>	<b>TIPO: Subvi</b>



**TÍTULO:** String de Datos RF de la Trama API de Recepción

**MÓDULO:** Transmisión y Recepción Serial API

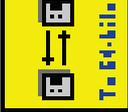
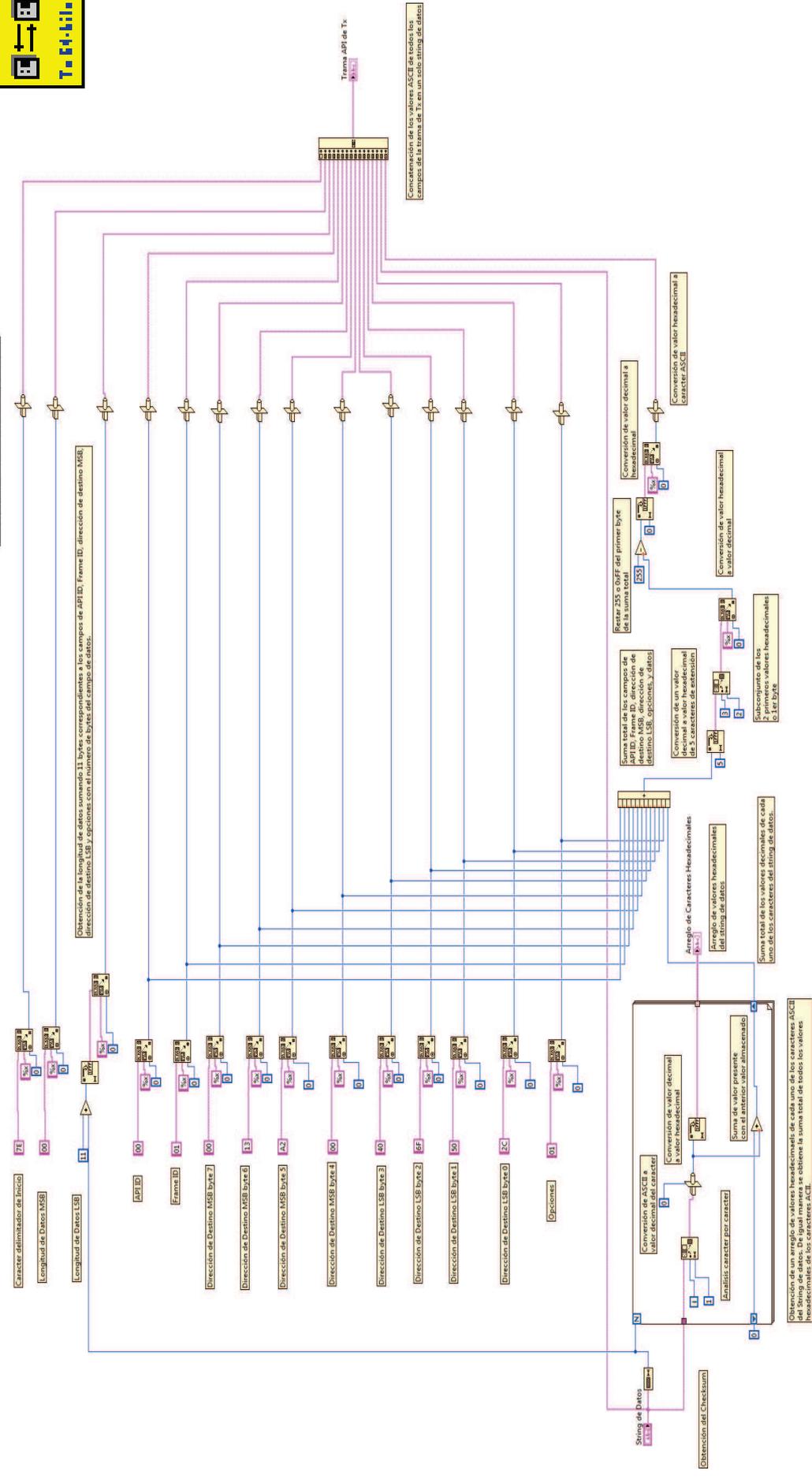
**ANEXO:** C1.4

**ESQUEMA No:** 5 de 6

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi

Conversión de valores hexadecimales a caracteres ASCII



**TÍTULO:** Trama API de Transmisión de 64-bits de Direccionamiento

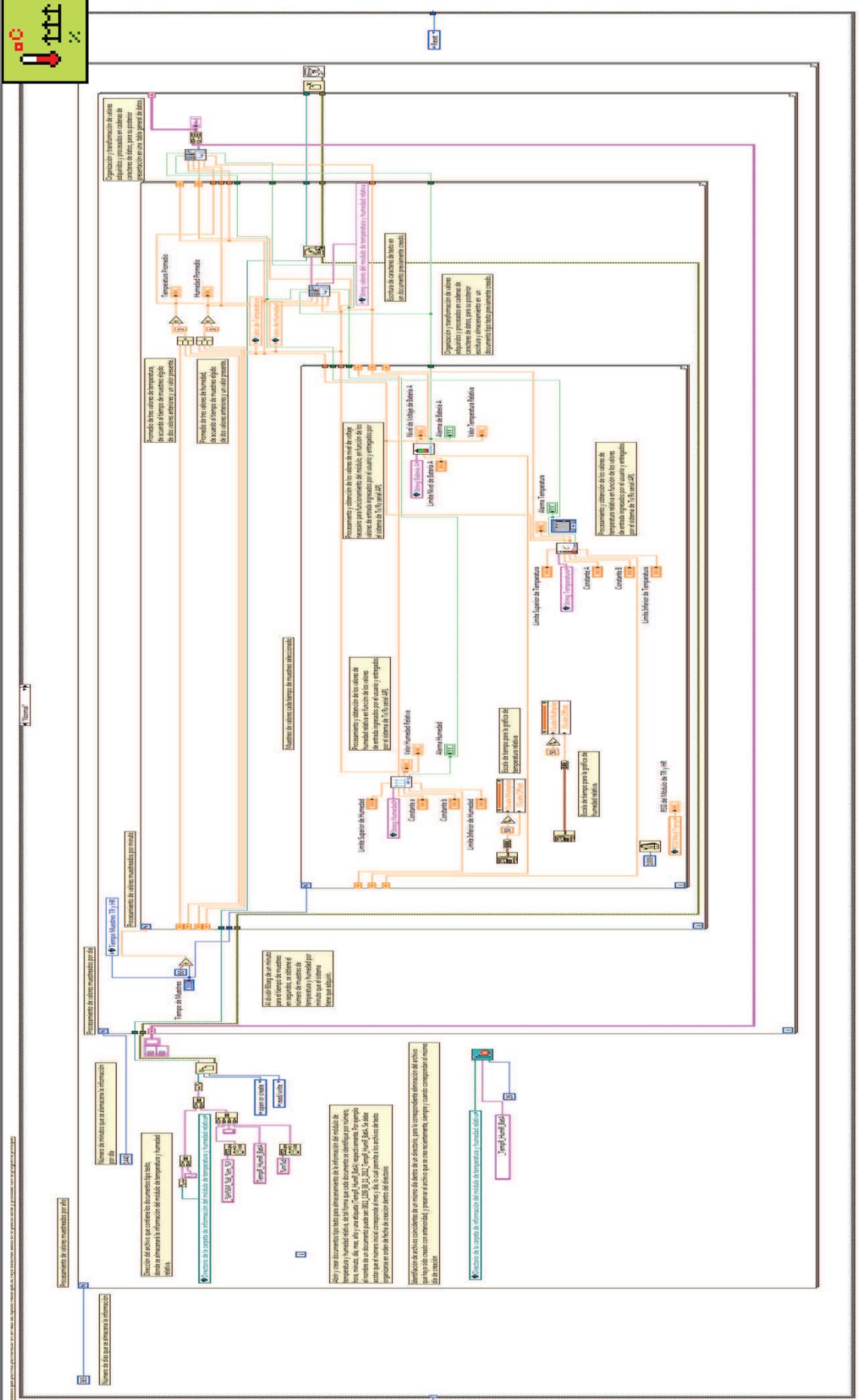
**MÓDULO:** Transmisión y Recepción Serial API

**ANEXO:** C1.5

**ESQUEMA No:** 6 de 6

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi



**TÍTULO:** Diagrama de Bloques Inicial de la Adquisición de Temperatura, Humedad y Nivel de Batería A

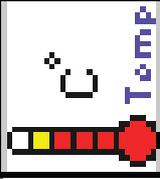
**MÓDULO:** Adquisición de Temperatura y Humedad Relativa

**ANEXO:** C2

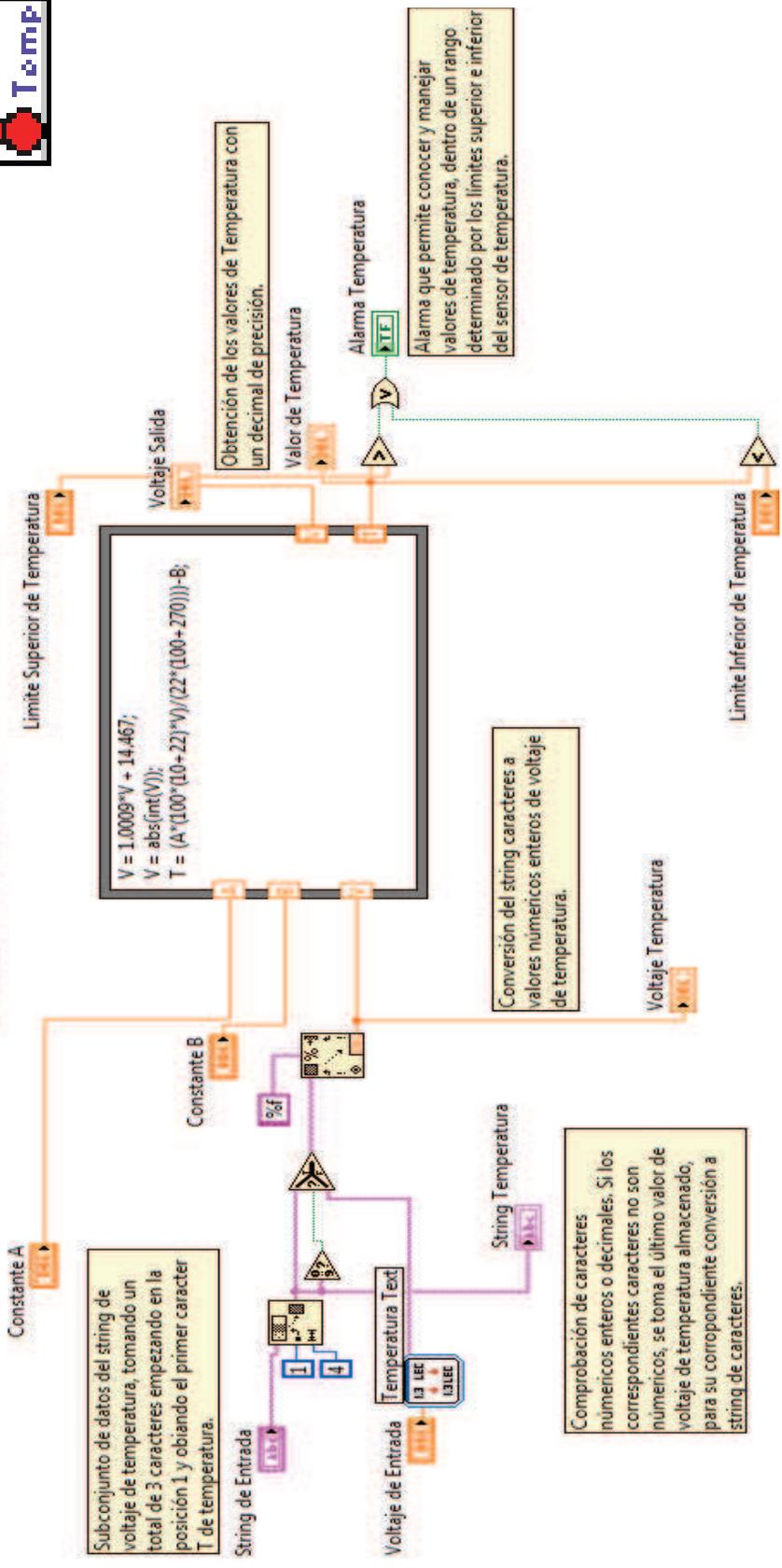
**ESQUEMA No:** 1 de 6

**NOMBRE:** Miguel Bonilla

**TIPO:** VI



Función lineal que permite obtener los valores de Temperatura, por medio de las constantes A y B y del valor entero de voltaje de temperatura obtenido.



**TÍTULO:** Valores de Temperatura Relativa

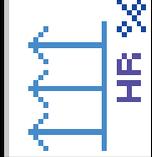
**MÓDULO:** Adquisición de Temperatura y Humedad Relativa

**ANEXO:** C2.1

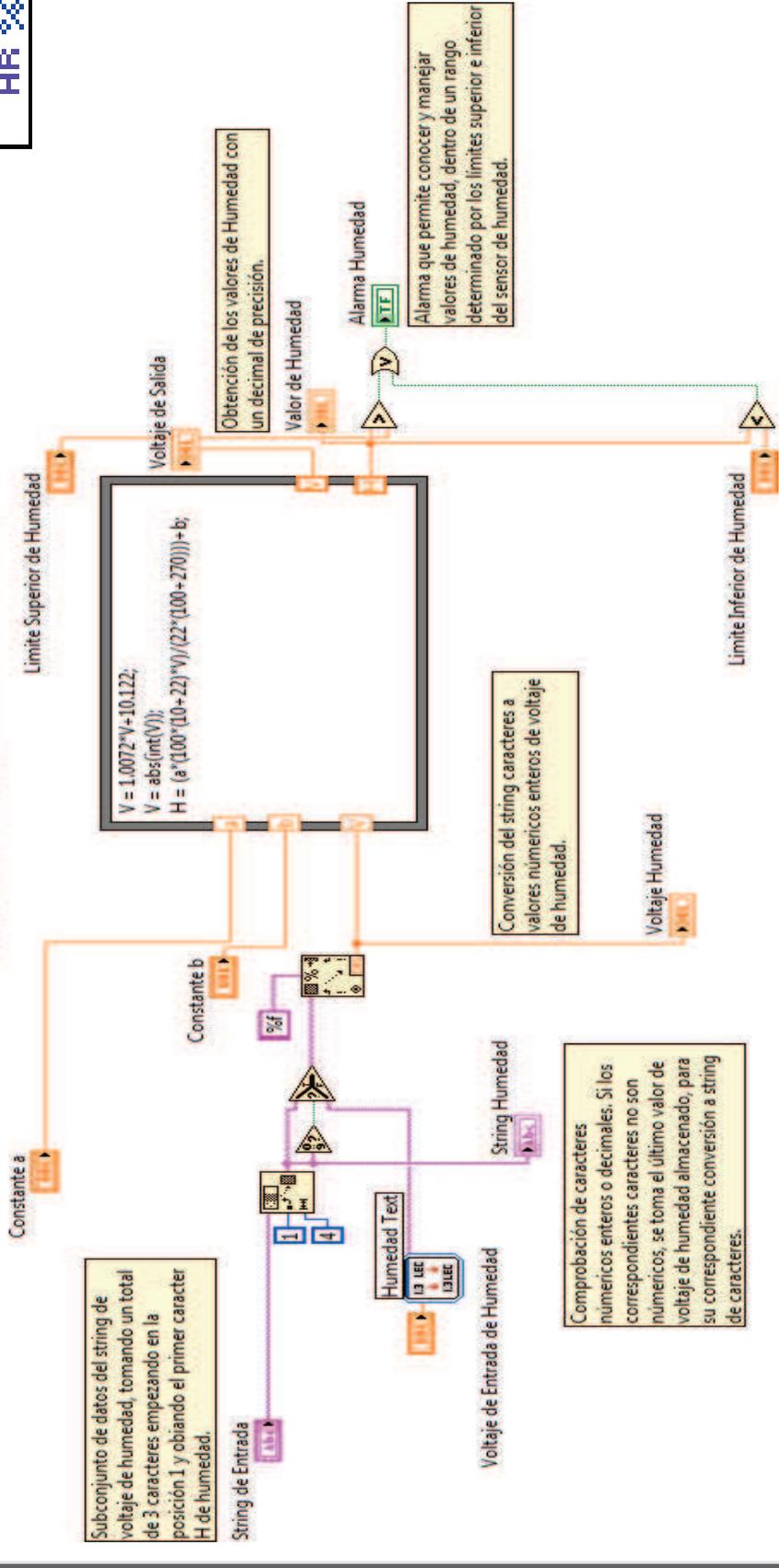
**ESQUEMA No:** 2 de 6

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi



Función lineal que permite obtener los valores de Humedad, por medio de las constantes A y B y del valor entero de voltaje de humedad obtenido.



**TÍTULO:** Valores de Humedad Relativa

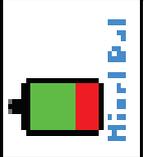
**MÓDULO:** Adquisición de Temperatura y Humedad Relativa

**ANEXO:** C2.2

**ESQUEMA No:** 3 de 6

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi



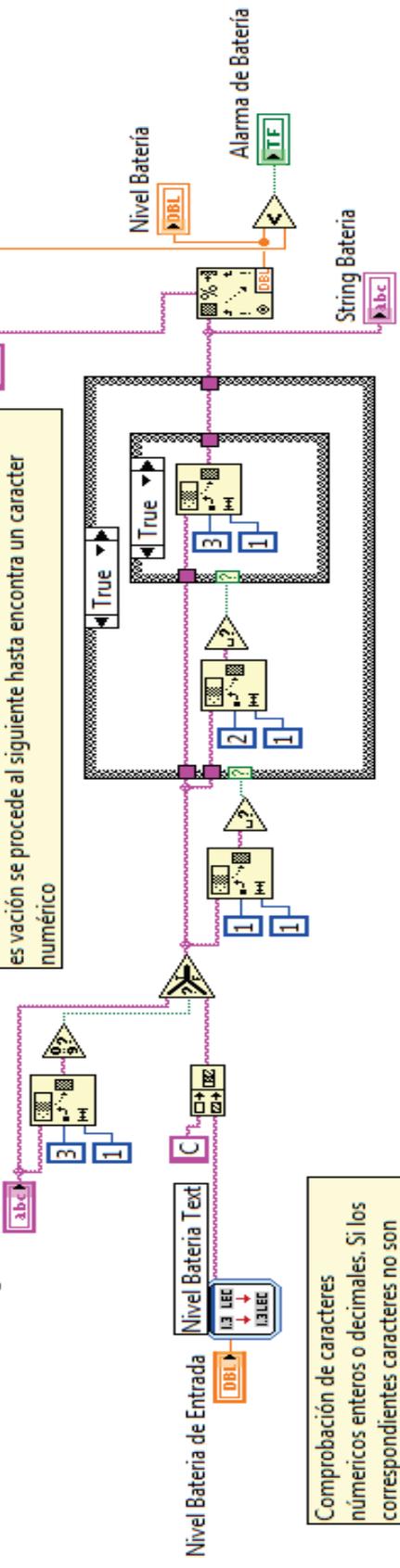
Alarma que permite conocer y manejar valores de nivel de batería hasta un determinado límite inferior.

Subconjunto de datos del string de valores de nivel de batería, tomando un total de 3 caracteres empezando en la posición 1 y obviando el primer carácter A o B de nivel de batería.

String de Entrada

Comprobación de caracteres vacíos, caso en el cual se procederá a detectar el siguiente carácter. Si el siguiente carácter también es vacío se procede al siguiente hasta encontrar un carácter numérico

Límite Nivel de Batería



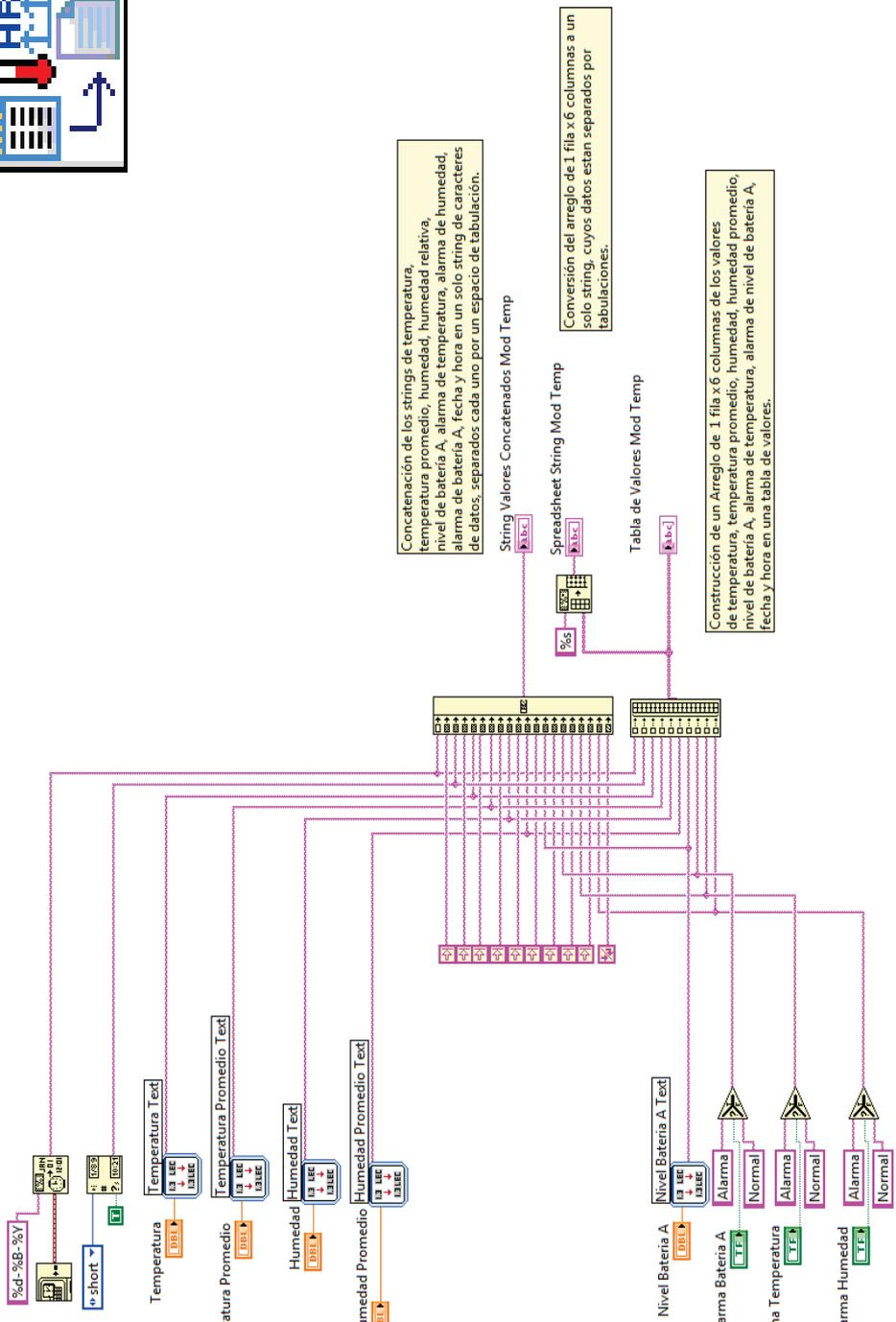
Comprobación de caracteres numéricos enteros o decimales. Si los correspondientes caracteres no son numéricos, se toma el último valor de nivel de batería almacenado, para su correspondiente conversión a string de caracteres.

Conversión del string de caracteres, a valores numéricos flotantes de nivel de batería, con ningún decimal de precisión.

<b>TÍTULO:</b> Valores de Nivel de Batería A o B		<b>MÓDULO:</b> Adquisición de Temperatura y Humedad Relativa	
<b>ANEXO:</b> C2.3	<b>ESQUEMA No:</b> 4 de 6	<b>NOMBRE:</b> Miguel Bonilla	<b>TIPO:</b> Subvi



Obtención de un string de fecha y hora de la adquisición de datos en: día-mes-año horas:segundos



Concatenación de los strings de temperatura, temperatura promedio, humedad, humedad relativa, nivel de batería A, alarma de temperatura, alarma de humedad, alarma de batería A, fecha y hora en un solo string de caracteres de datos, separados cada uno por un espacio de tabulación.

Conversión de Valores Flotantes de:  
 - Temperatura,  
 - Temperatura Promedio,  
 - Humedad,  
 - Humedad Promedio,  
 - Nivel de Batería A  
 a string de caracteres

String Valores Concatenados Mod Temp

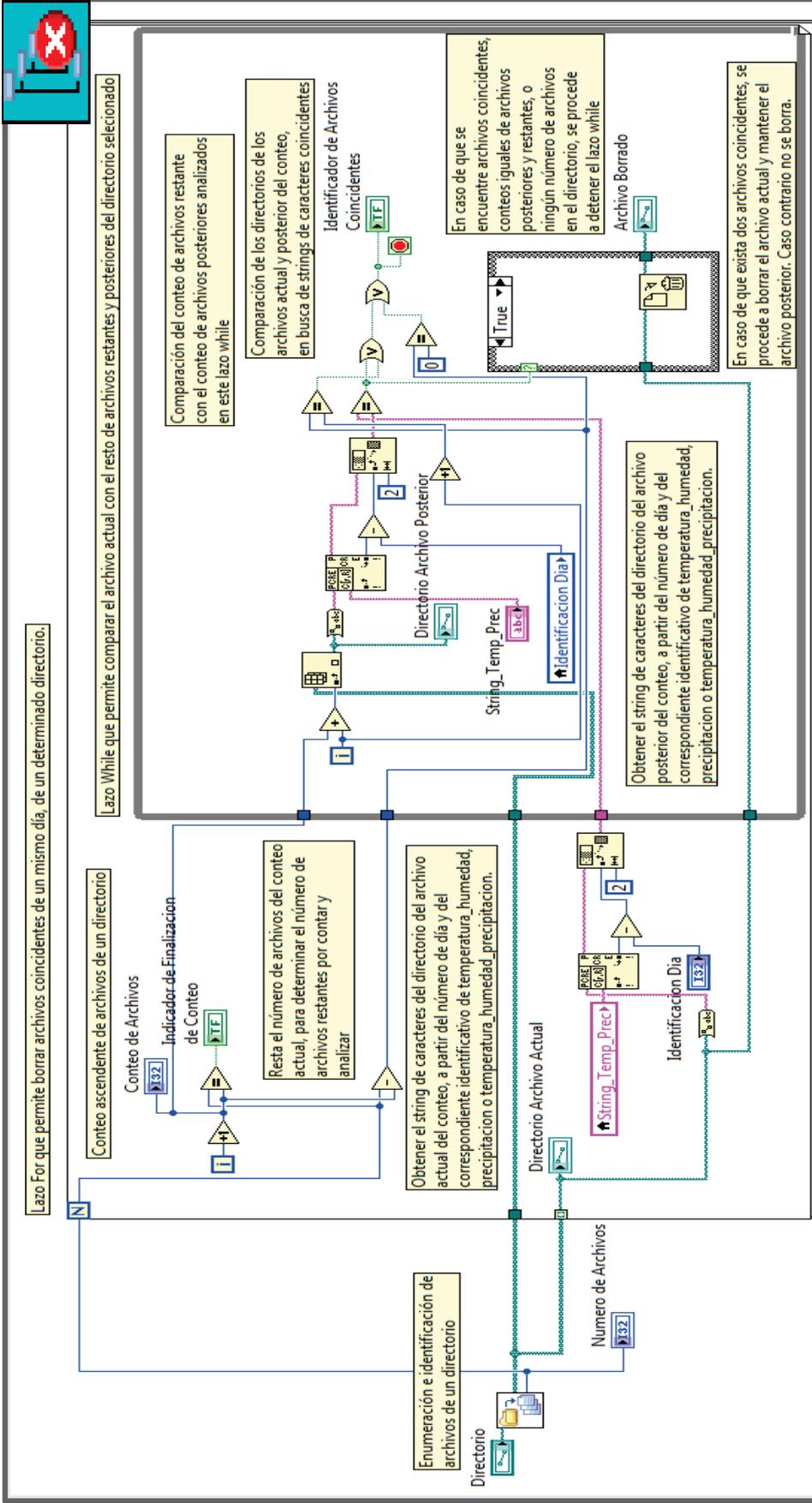
Spreadsheet String Mod Temp

Tabla de Valores Mod Temp

Construcción de un Arreglo de 1 fila x 6 columnas de los valores de temperatura, temperatura promedio, humedad, humedad promedio, nivel de batería A, alarma de temperatura, alarma de nivel de batería A, fecha y hora en una tabla de valores.

Detección de Alarmas de Temperatura, de Humedad y Nivel de Batería A como Normal o Alarmada de acuerdo a los límites de trabajo del sensor de Temperatura y del Nivel de Batería A

<b>TÍTULO: Tabla de Valores de Temperatura, Humedad y Batería A</b>		<b>MÓDULO: Adquisición de Temperatura y Humedad Relativa</b>	
<b>ANEXO: C2.4</b>	<b>ESQUEMA No: 5 de 6</b>	<b>NOMBRE: Miguel Bonilla</b>	<b>TIPO: Subvi</b>



**TÍTULO:** Eliminación de Archivos Repetidos por Día

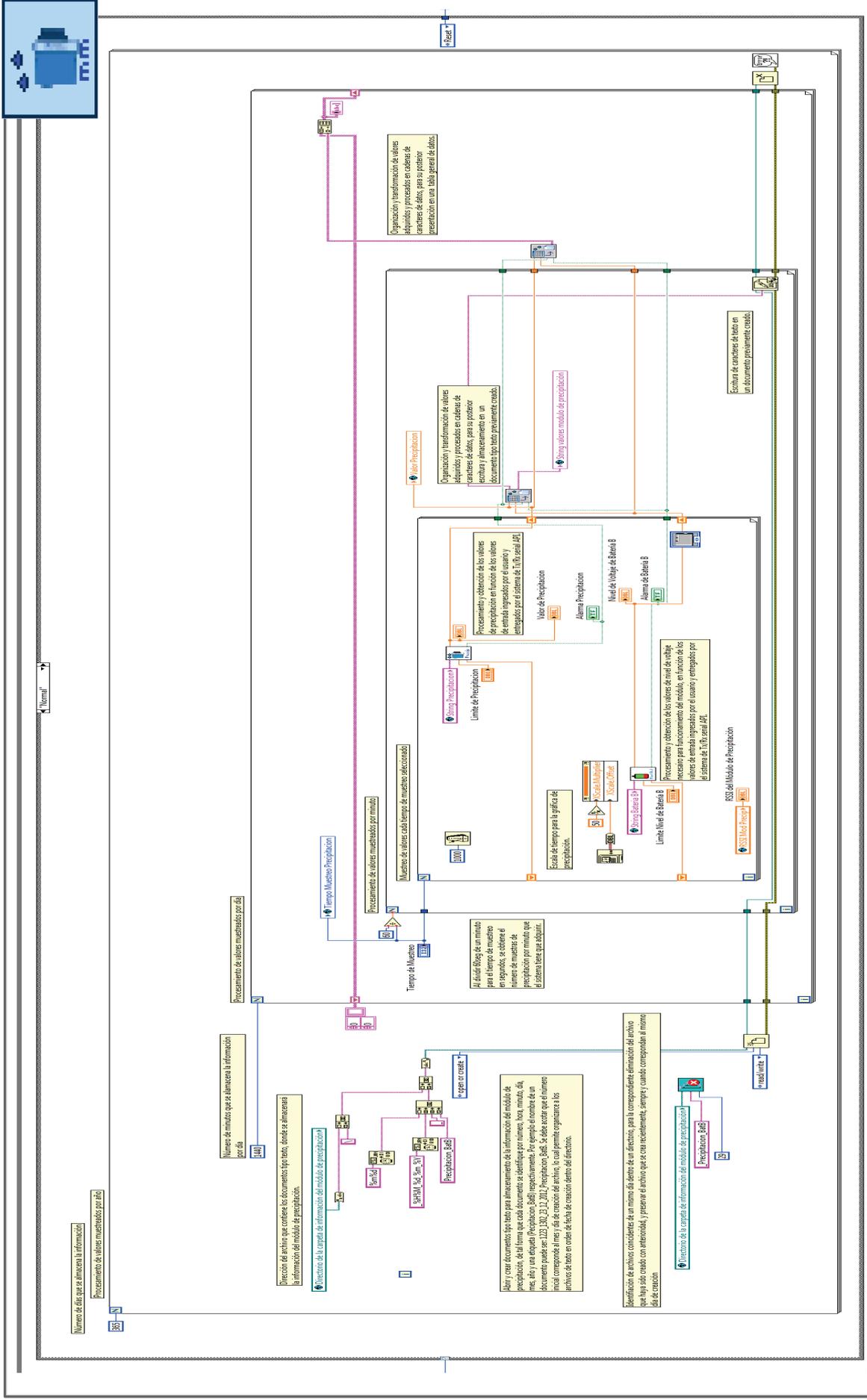
**MÓDULO:** Adquisición de Temperatura y Humedad Relativa

**ANEXO:** C2.5

**ESQUEMA No:** 6 de 6

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi



MÓDULO: Adquisición de Precipitación

ANEXO: C3

NOMBRE: Miguel Bonilla

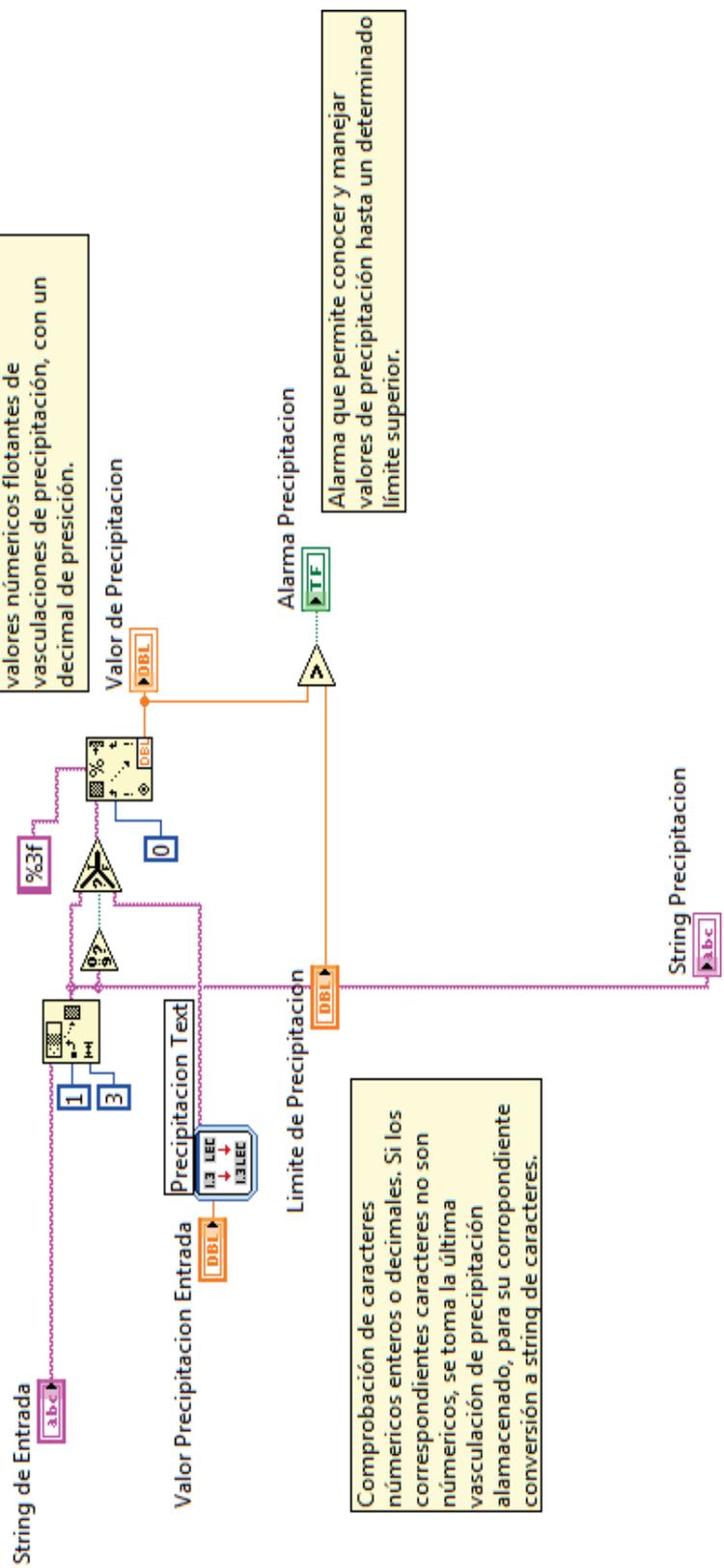
TIPO: VI

1 de 3

ESQUEMA No:



Subconjunto de datos del string de vasculaciones de precipitación, tomando un total de 3 caracteres empezando en la posición 1 y obviando el primer caracter P de precipitación.



Conversión del string de caracteres, a valores numéricos flotantes de vasculaciones de precipitación, con un decimal de precisión.

Alarma que permite conocer y manejar valores de precipitación hasta un determinado límite superior.

Comprobación de caracteres numéricos enteros o decimales. Si los correspondientes caracteres no son numéricos, se toma la última vasculación de precipitación almacenado, para su correspondiente conversión a string de caracteres.

**TÍTULO:** Valores de Precipitación

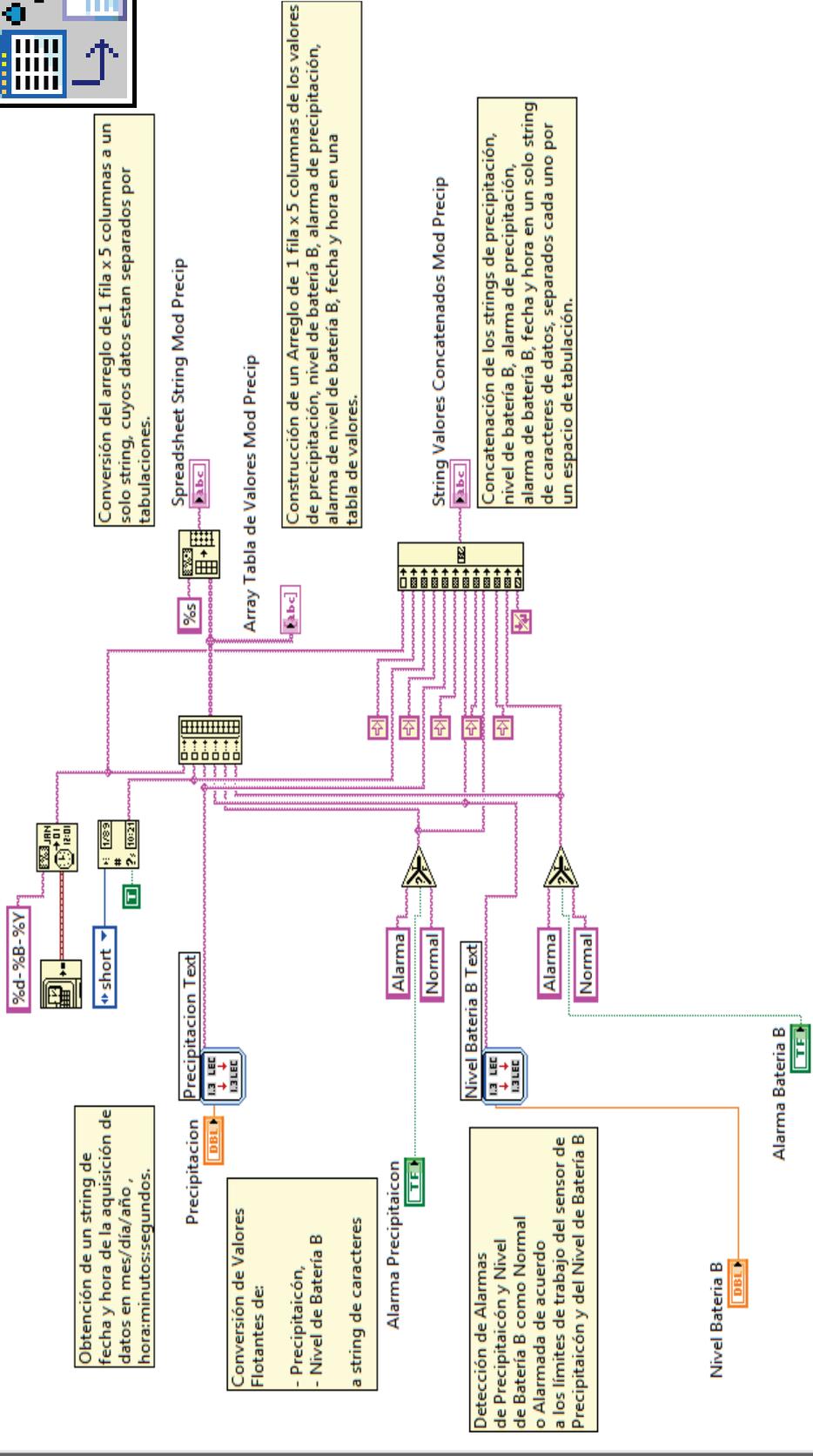
**MÓDULO:** Adquisición de Precipitación

**ANEXO:** C3.1

**ESQUEMA No:** 2 de 3

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi



**TÍTULO:** Tabla de Valores de Precipitación y Batería B

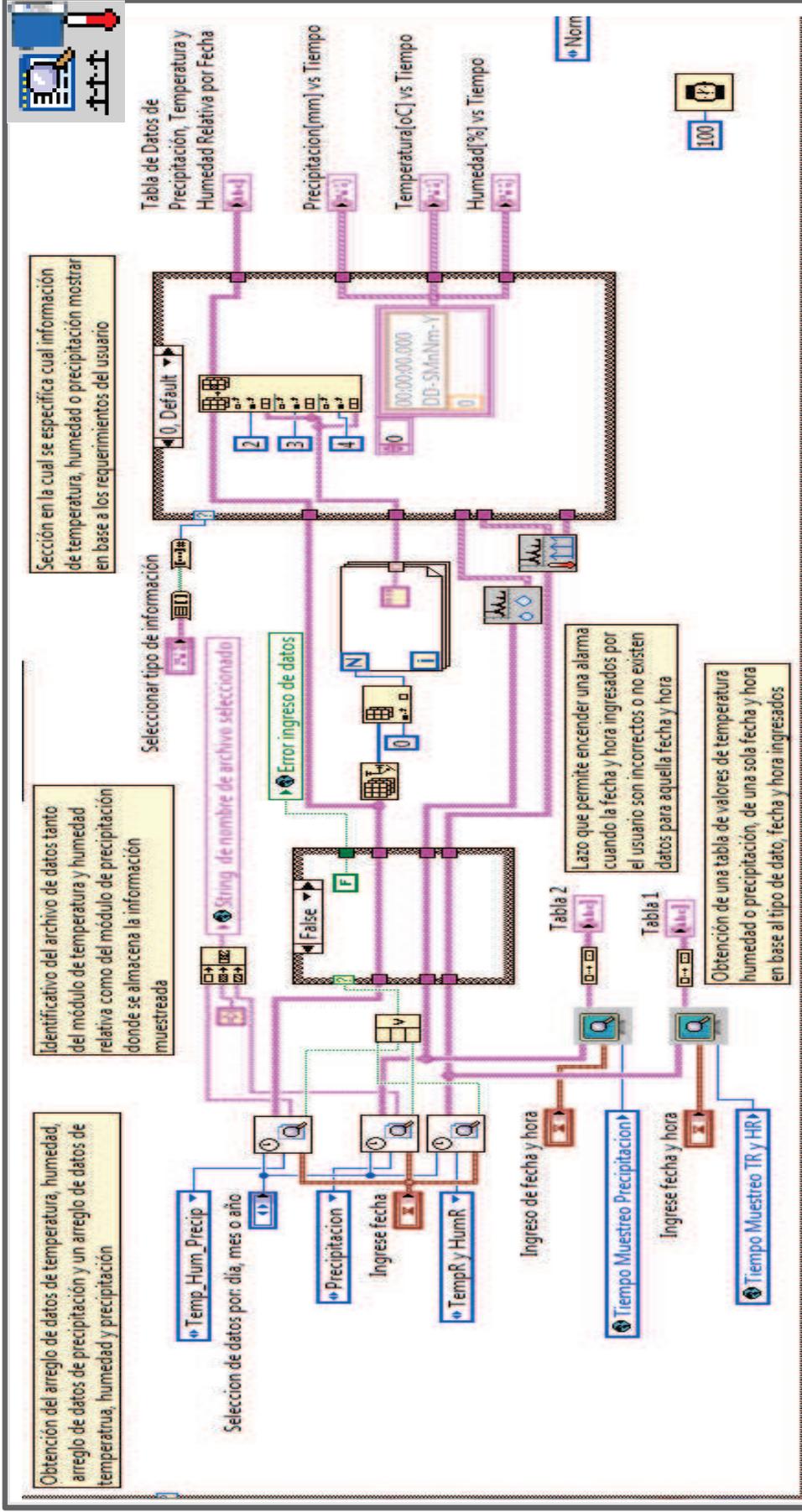
**MÓDULO:** Adquisición de Precipitación

**ANEXO:** C3.2

**ESQUEMA No:** 3 de 3

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi



**TÍTULO:** Diagrama de Bloques Inicial de la Presentación de Datos Meteorológicos

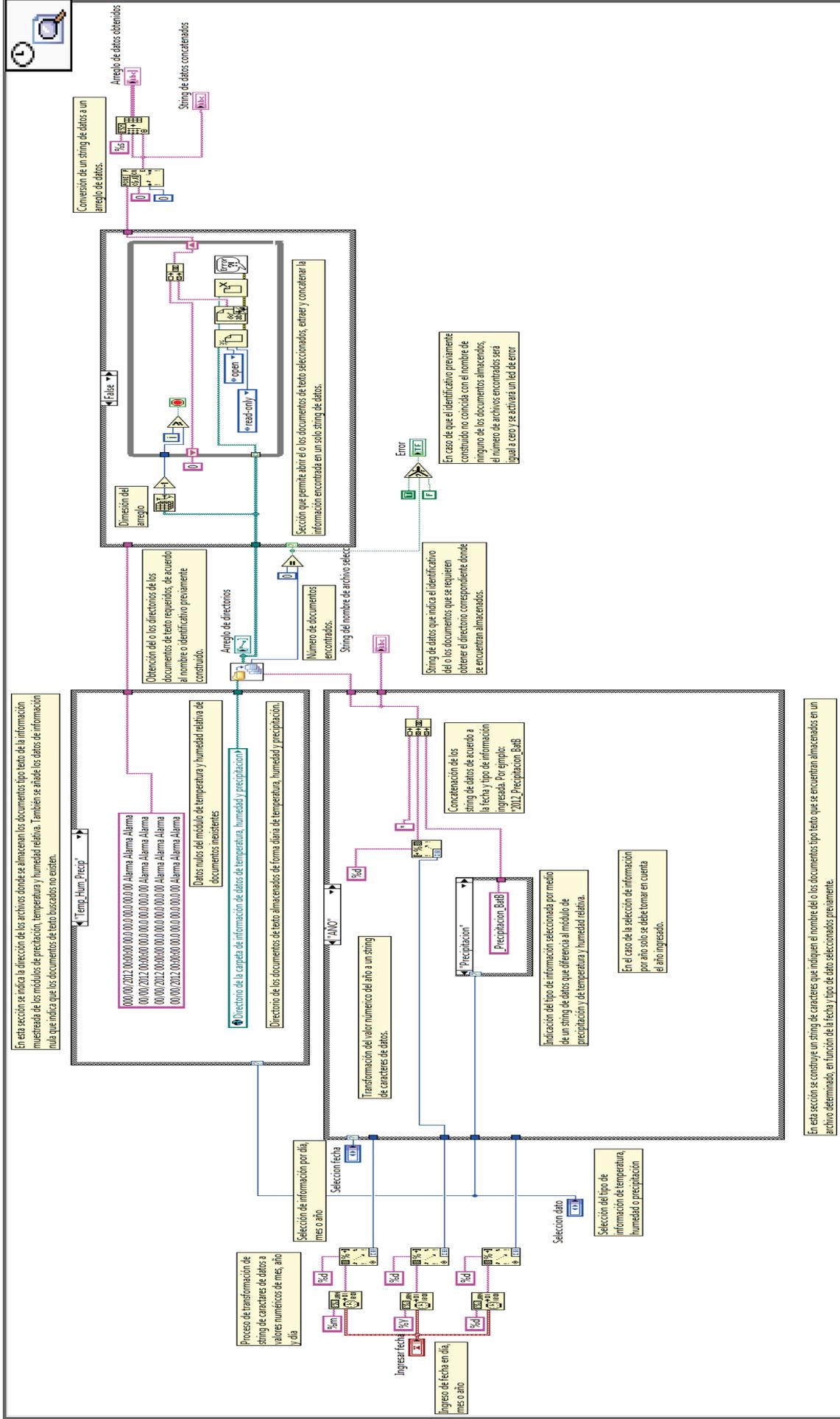
**MÓDULO:** Información Meteorológica

**ANEXO:** C4

**ESQUEMA No:** 1 de 5

**NOMBRE:** Miguel Bonilla

**TIPO:** VI



**TÍTULO:** Concatenación de Datos de un Archivo de Texto Seleccionado

**MÓDULO:** Información Meteorológica

En esta sección se construye un string de caracteres que indiquen el nombre de los documentos tipo texto que se encuentran almacenados en un archivo determinado, en función de la fecha y tipo de dato seleccionados previamente.

**ANEXO:** C4.1

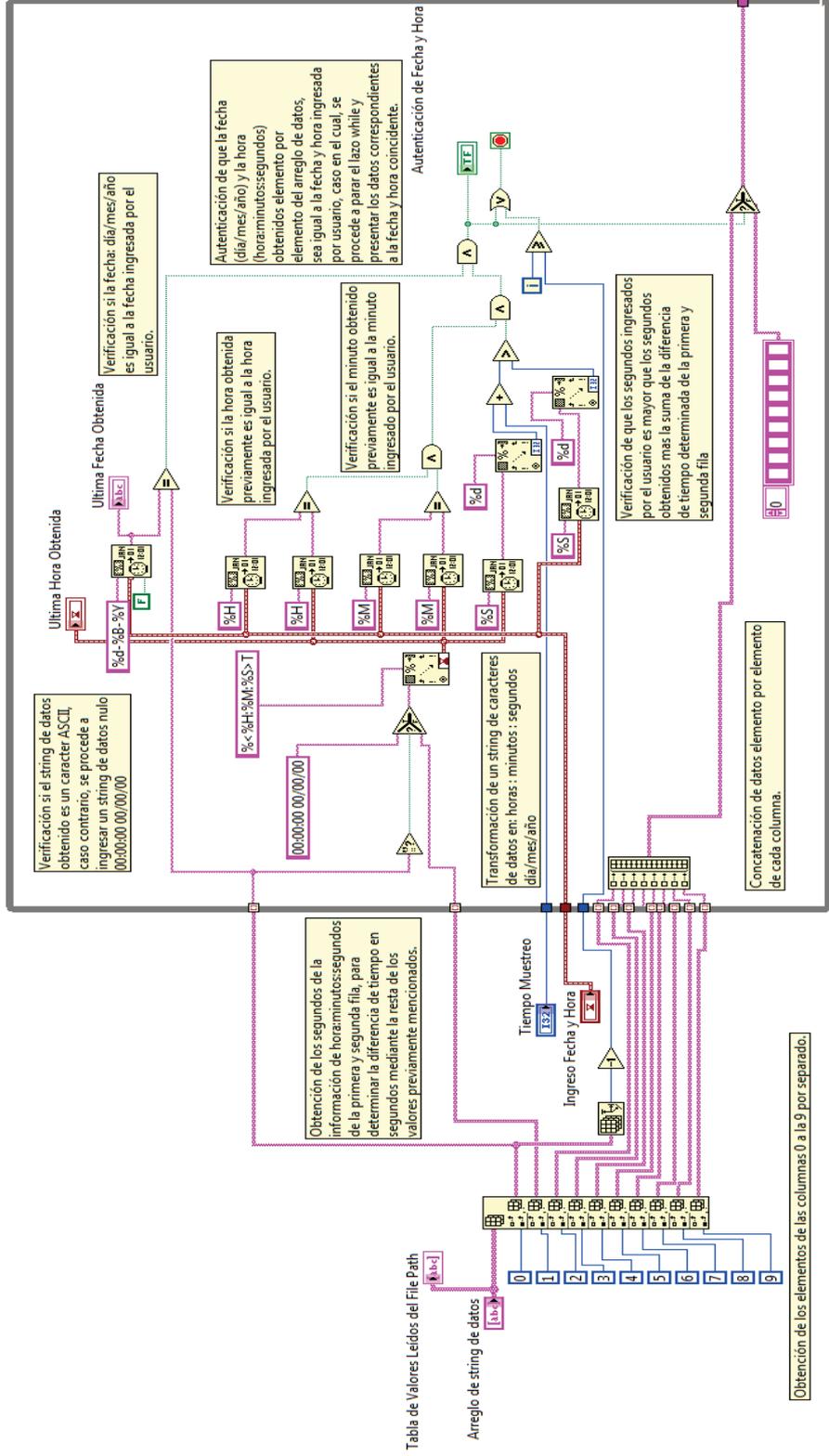
**ESQUEMA No:** 2 de 5

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi



Sección en la cual se obtiene la información, ya sea del módulo de precipitación o temperatura y humedad relativa, por medio de la autenticación de la fecha y hora ingresada por el usuario y que coincida con la fecha y hora obtenida del arreglo de datos.



**TÍTULO:** Obtención de Información Meteorológica Correspondiente a una Sola Fecha Especifica

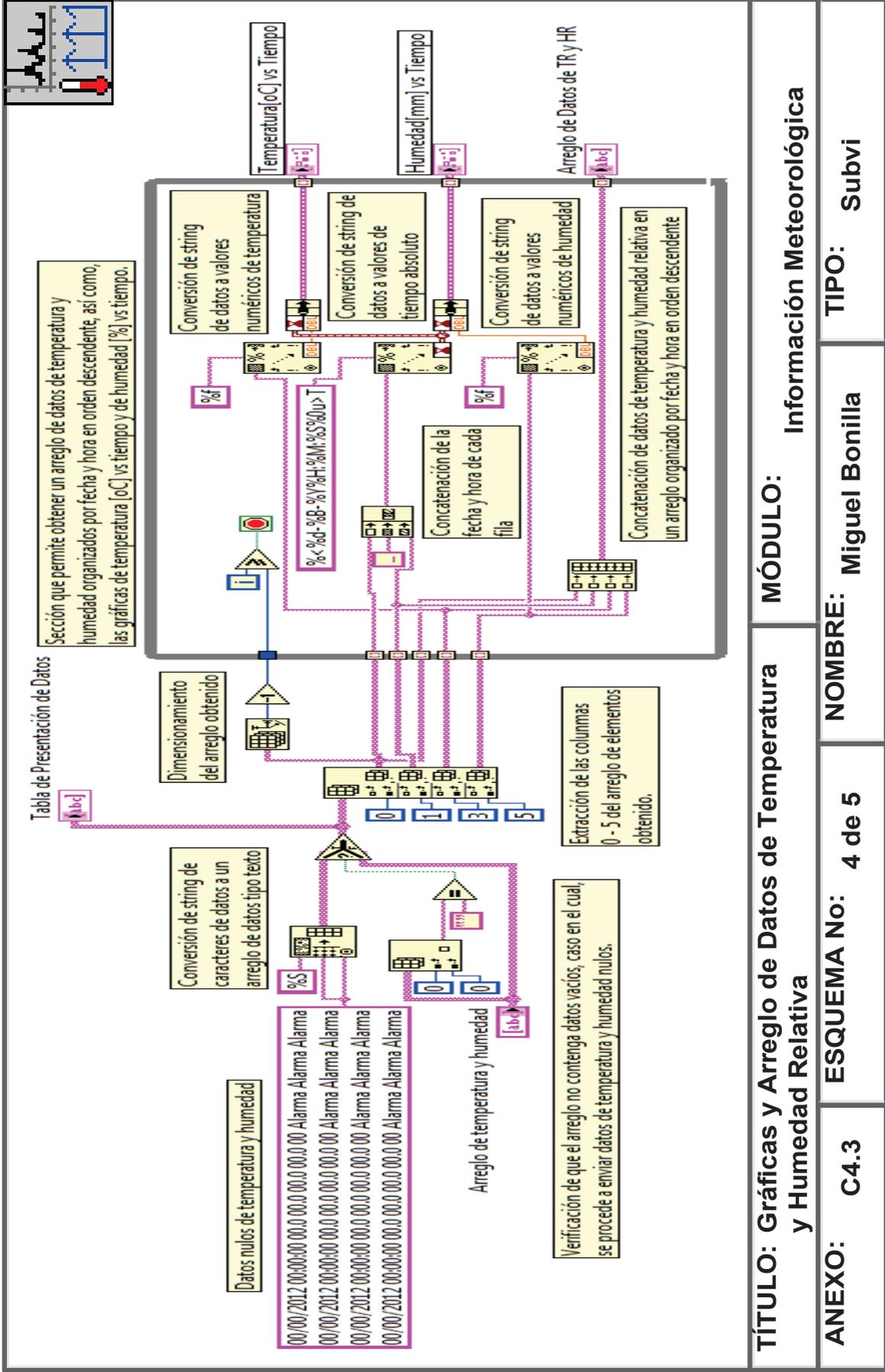
**MÓDULO:** Información Meteorológica

**ANEXO:** C4.2

**ESQUEMA No:** 3 de 5

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi



**TÍTULO: Gráficas y Arreglo de Datos de Temperatura y Humedad Relativa**

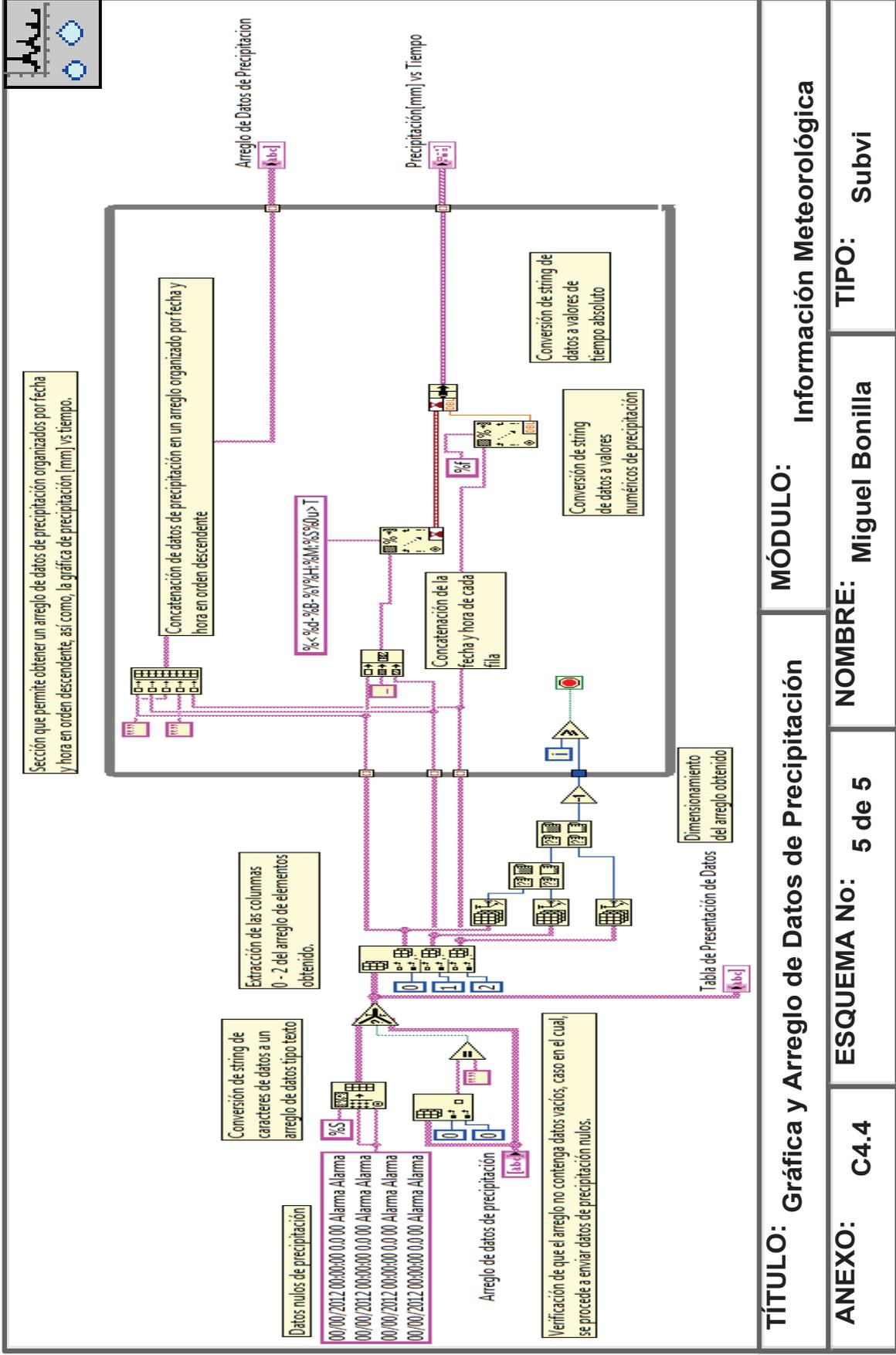
**MÓDULO: Información Meteorológica**

**ANEXO: C4.3**

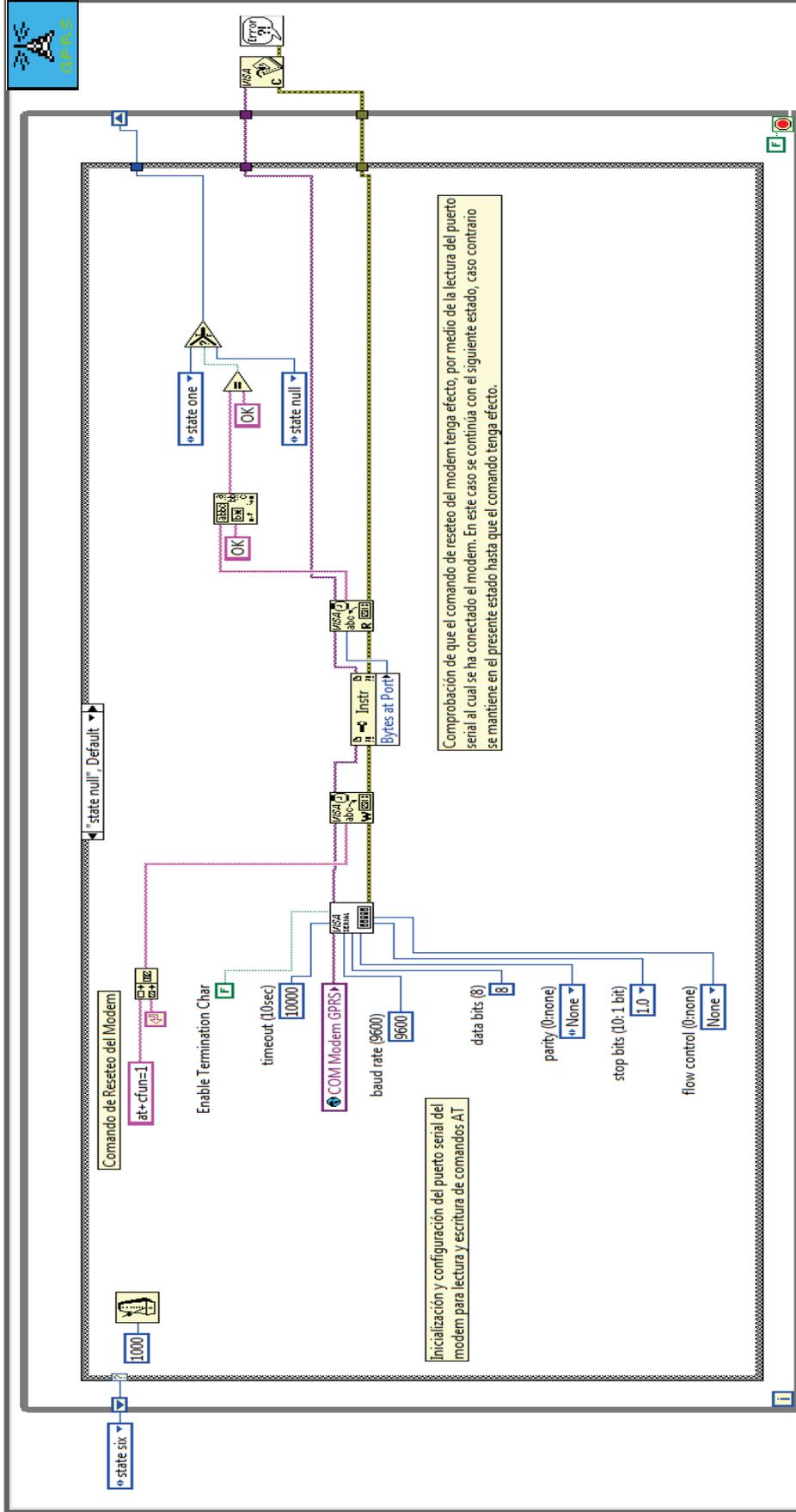
**ESQUEMA No: 4 de 5**

**NOMBRE: Miguel Bonilla**

**TIPO: Subvi**



<b>TÍTULO:</b> Gráfica y Arreglo de Datos de Precipitación	<b>MÓDULO:</b> Información Meteorológica
<b>ANEXO:</b> C4.4	<b>NOMBRE:</b> Miguel Bonilla
<b>ESQUEMA No:</b> 5 de 5	<b>TIPO:</b> Subvi



Maquina de estados que permite la autentificación y asociación del modem dentro de la red GPRS, para a continuación establecer una comunicación orientada a la conexión y enviar la información de temperatura, humedad y precipitación cada 5 minutos.

**TÍTULO: Estado 0 de la Máquina de Estados de Transmisión GPRS**

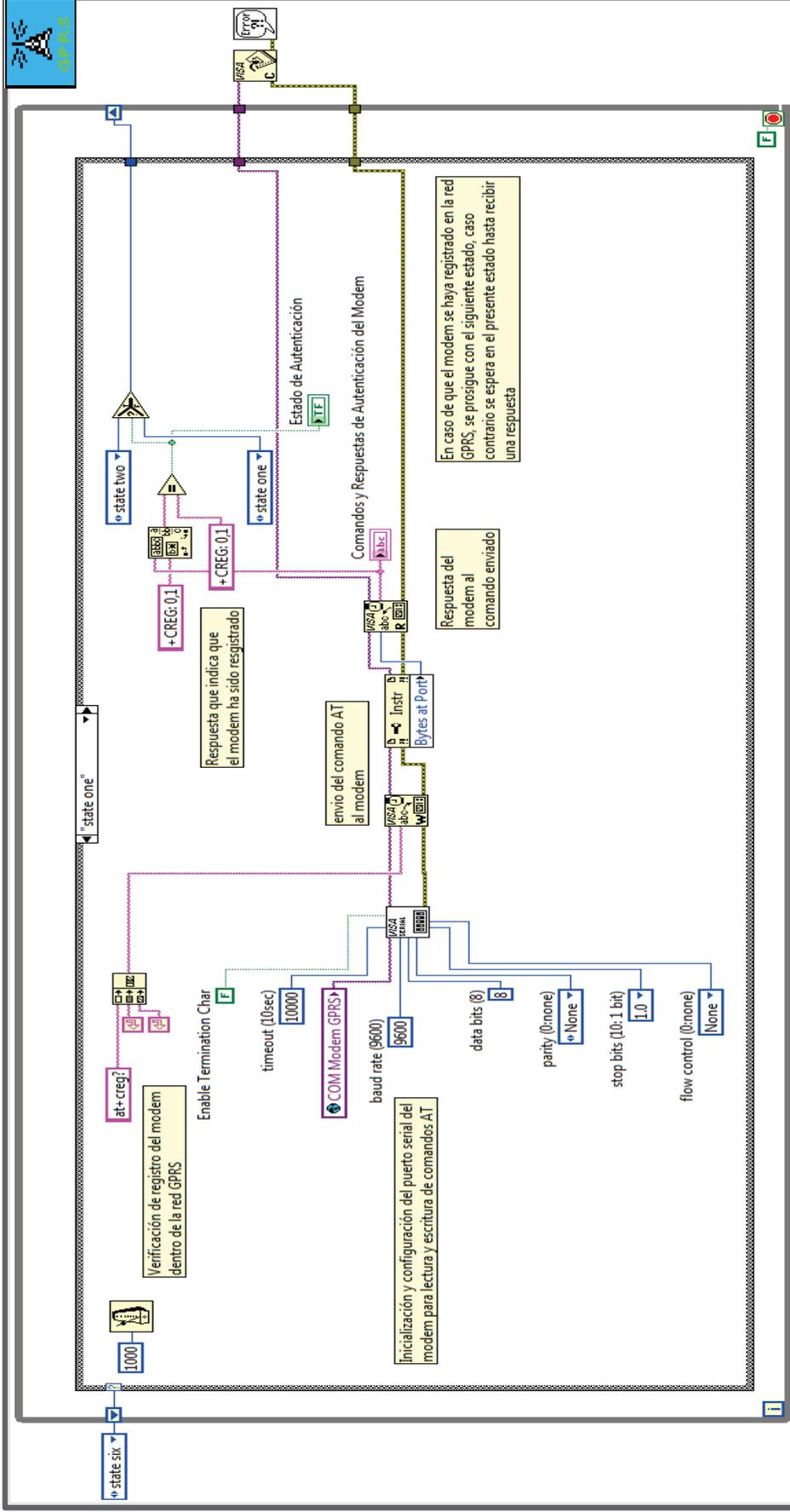
**MÓDULO: Transmisión de Datos GPRS**

**ANEXO: C5**

**ESQUEMA No: 1 de 8**

**NOMBRE: Miguel Bonilla**

**TIPO: VI**



Maquina de estados que permite la autentificación y asociación del modem dentro de la red GPRS, para a continuación establecer una comunicación orientada a la conexión y enviar la información de temperatura, humedad y precipitación cada 5 minutos.

**TÍTULO: Estado 1 de la Máquina de Estados de Transmisión GPRS**

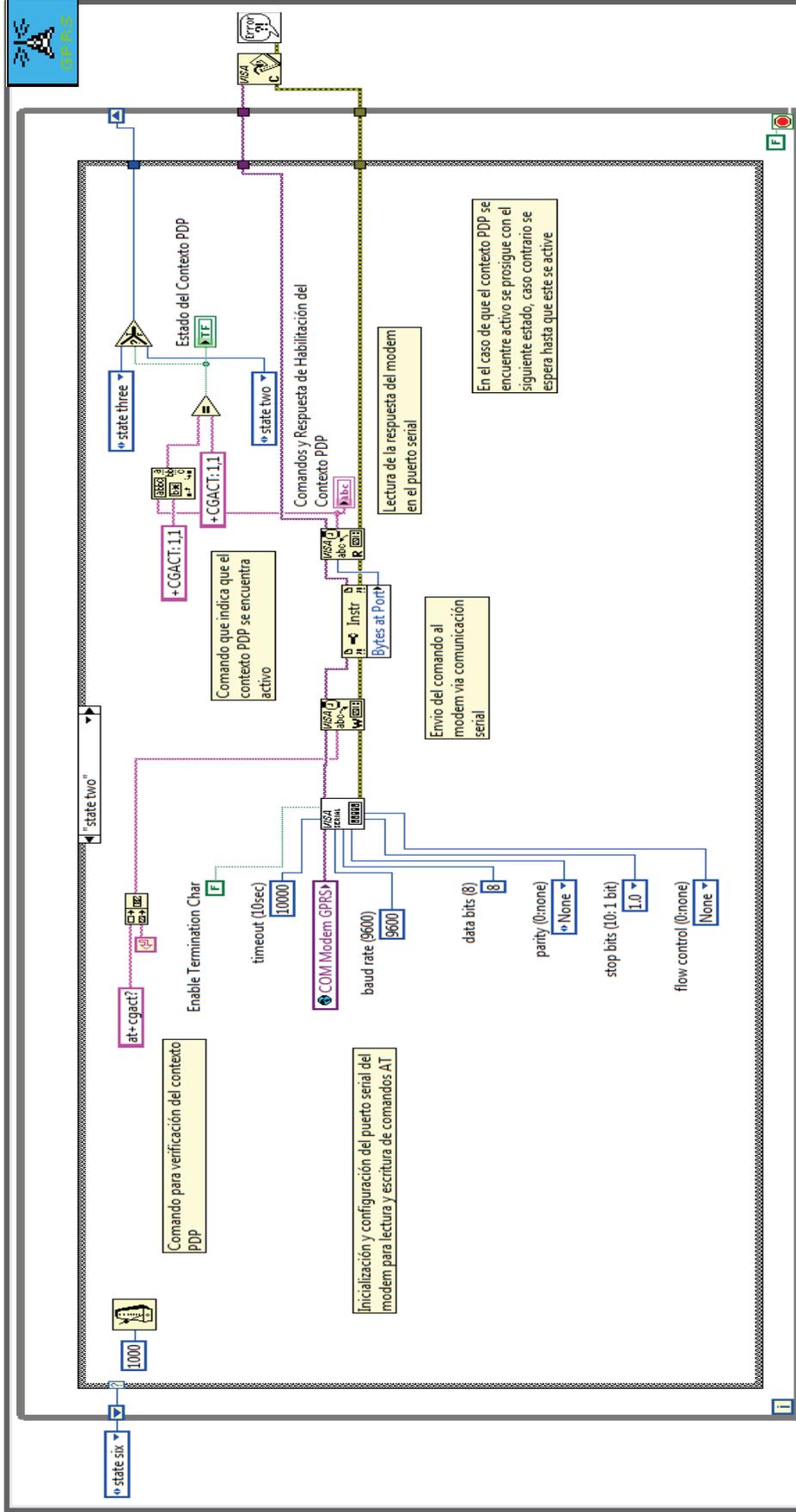
**MÓDULO: Transmisión de Datos GPRS**

**ANEXO: C5.1**

**ESQUEMA No: 2 de 8**

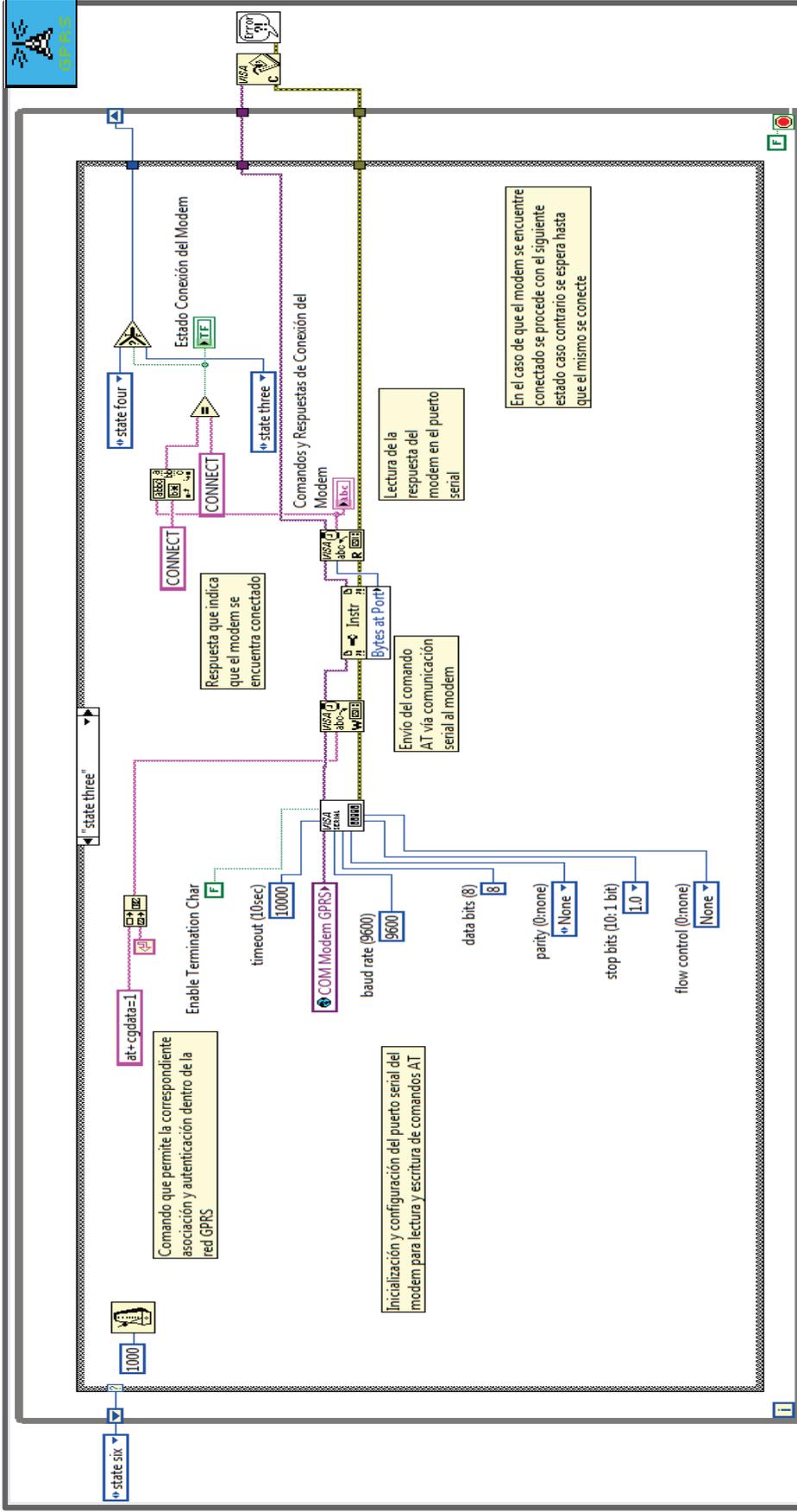
**NOMBRE: Miguel Bonilla**

**TIPO: Vi**



Maquina de estados que permite la autenticación y asociación del modem dentro de la red GPRS, para a continuación establecer una comunicación orientada a la conexión y enviar la información de temperatura, humedad y precipitación cada 5 minutos.

<b>TÍTULO: Estado 2 de la Máquina de Estados de Transmisión GPRS</b>	<b>MÓDULO: Transmisión de Datos GPRS</b>
<b>ANEXO: C5.2</b>	<b>NOMBRE: Miguel Bonilla</b>
<b>ESQUEMA No: 3 de 8</b>	<b>TIPO: VI</b>



Maquina de estados que permite la autentificación y asociación del modem dentro de la red GPRS, para a continuación establecer una comunicación orientada a la conexión y enviar la información de temperatura, humedad y precipitación cada 5 minutos.

**TÍTULO: Estado 3 de la Máquina de Estados de Transmisión GPRS**

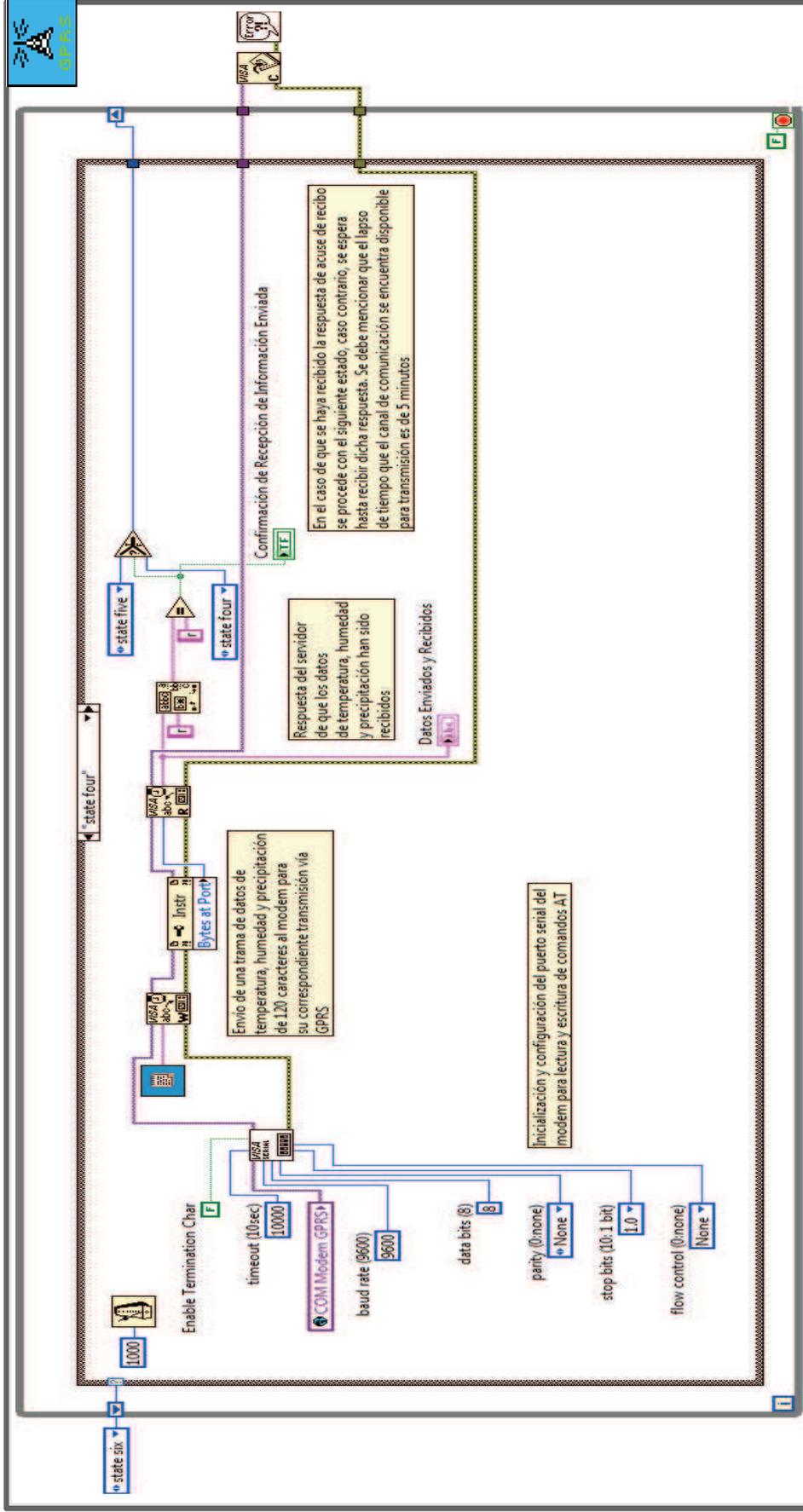
**MÓDULO: Transmisión de Datos GPRS**

**ANEXO: C5.3**

**ESQUEMA No: 4 de 8**

**NOMBRE: Miguel Bonilla**

**TIPO: Vi**



Maquina de estados que permite la autentificación y asociación del modem dentro de la red GPRS, para a continuación establecer una comunicación orientada a la conexión y enviar la información de temperatura, humedad y precipitación cada 5 minutos.

**TÍTULO:** Estado 4 de la Máquina de Estados de Transmisión GPRS

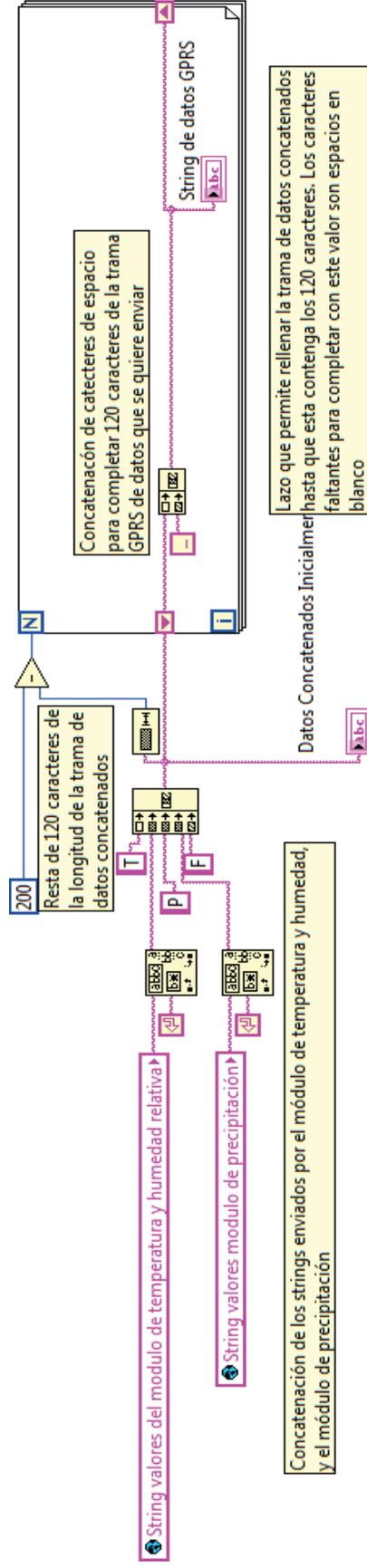
**MÓDULO:** Transmisión de Datos GPRS

**ANEXO:** C5.4

**ESQUEMA No:** 5 de 8

**NOMBRE:** Miguel Bonilla

**TIPO:** Vi



**TÍTULO:** Concatenación y Formación de Tramas de Datos GPRS

**MÓDULO:** Transmisión de Datos GPRS

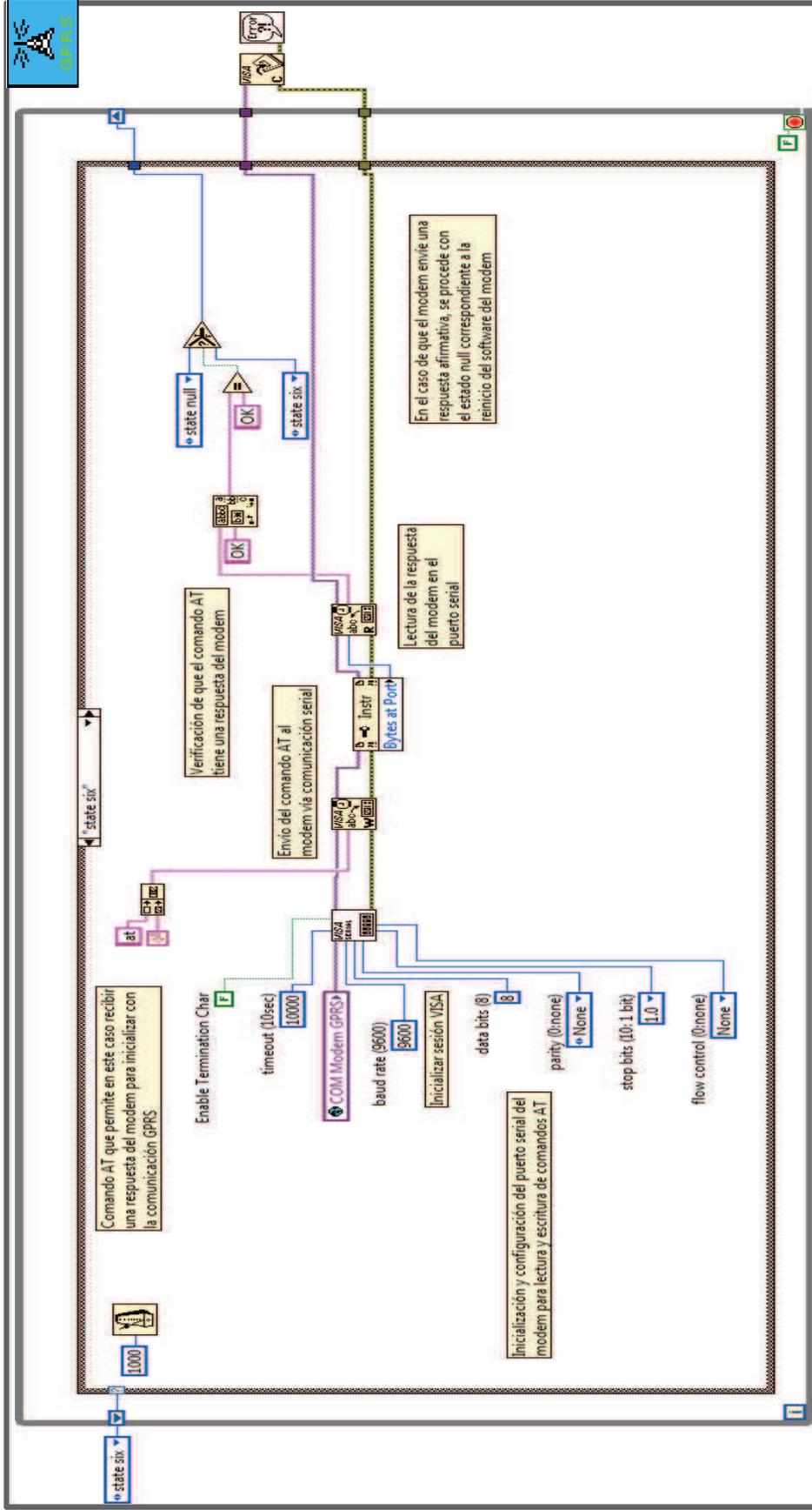
**ANEXO:** C5.5

**ESQUEMA No:** 6 de 8

**NOMBRE:** Miguel Bonilla

**TIPO:** Subvi

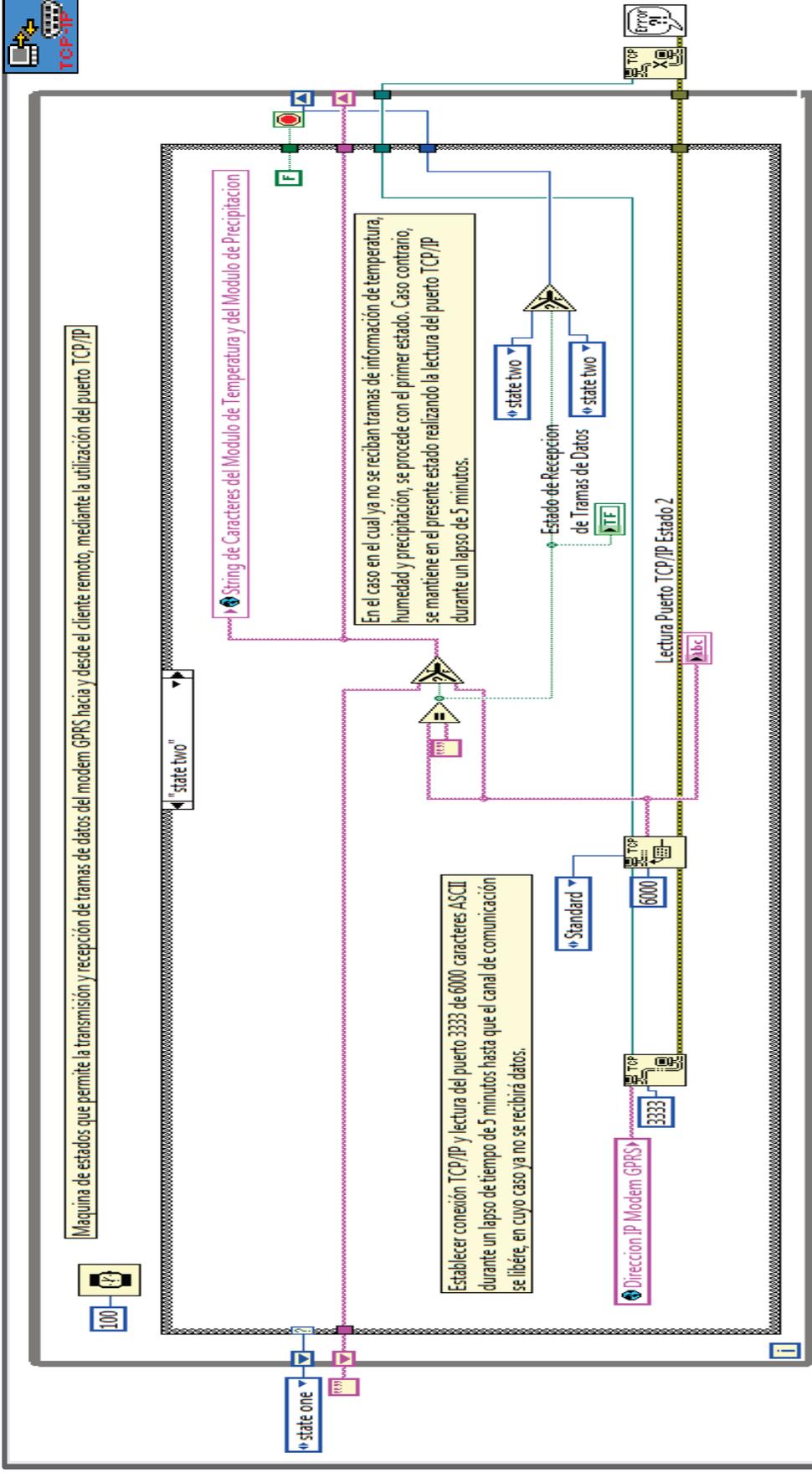




Máquina de estados que permite la autenticación y asociación del modem dentro de la red GPRS, para a continuación establecer una comunicación orientada a la conexión y enviar la información de temperatura, humedad y precipitación cada 5 minutos.

<b>TÍTULO: Estado 6 de la Máquina de Estados de Transmisión GPRS</b>		<b>MÓDULO: Transmisión de Datos GPRS</b>
<b>ANEXO: C5.7</b>	<b>ESQUEMA No: 8 de 8</b>	<b>NOMBRE: Miguel Bonilla</b>
		<b>TIPO: Vi</b>





**TÍTULO:** Estado 2 de la Máquina de Estados de Recepción GPRS

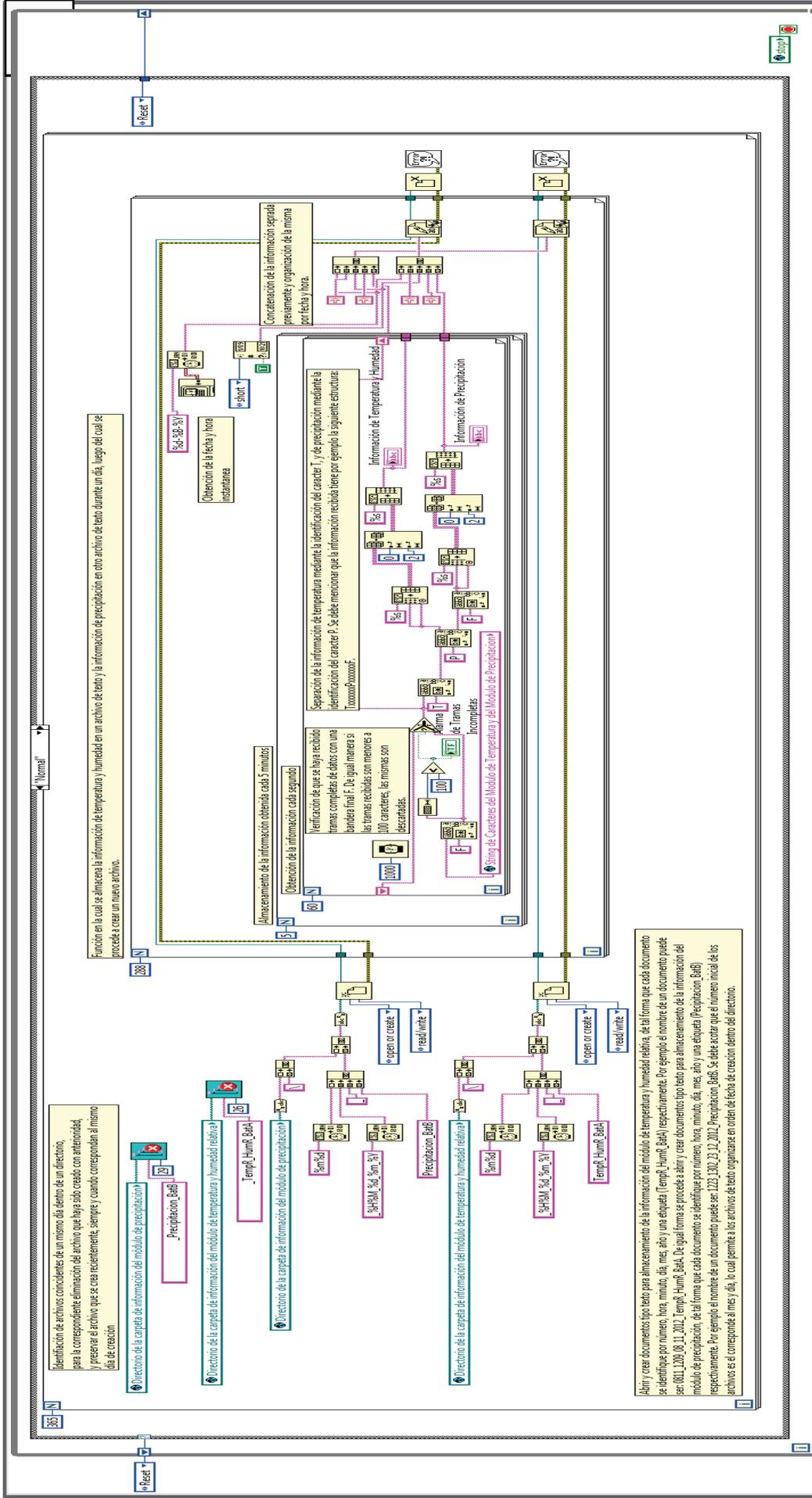
**MÓDULO:** Recepción de Datos GPRS

**ANEXO:** C6.1

**ESQUEMA No:** 2 de 2

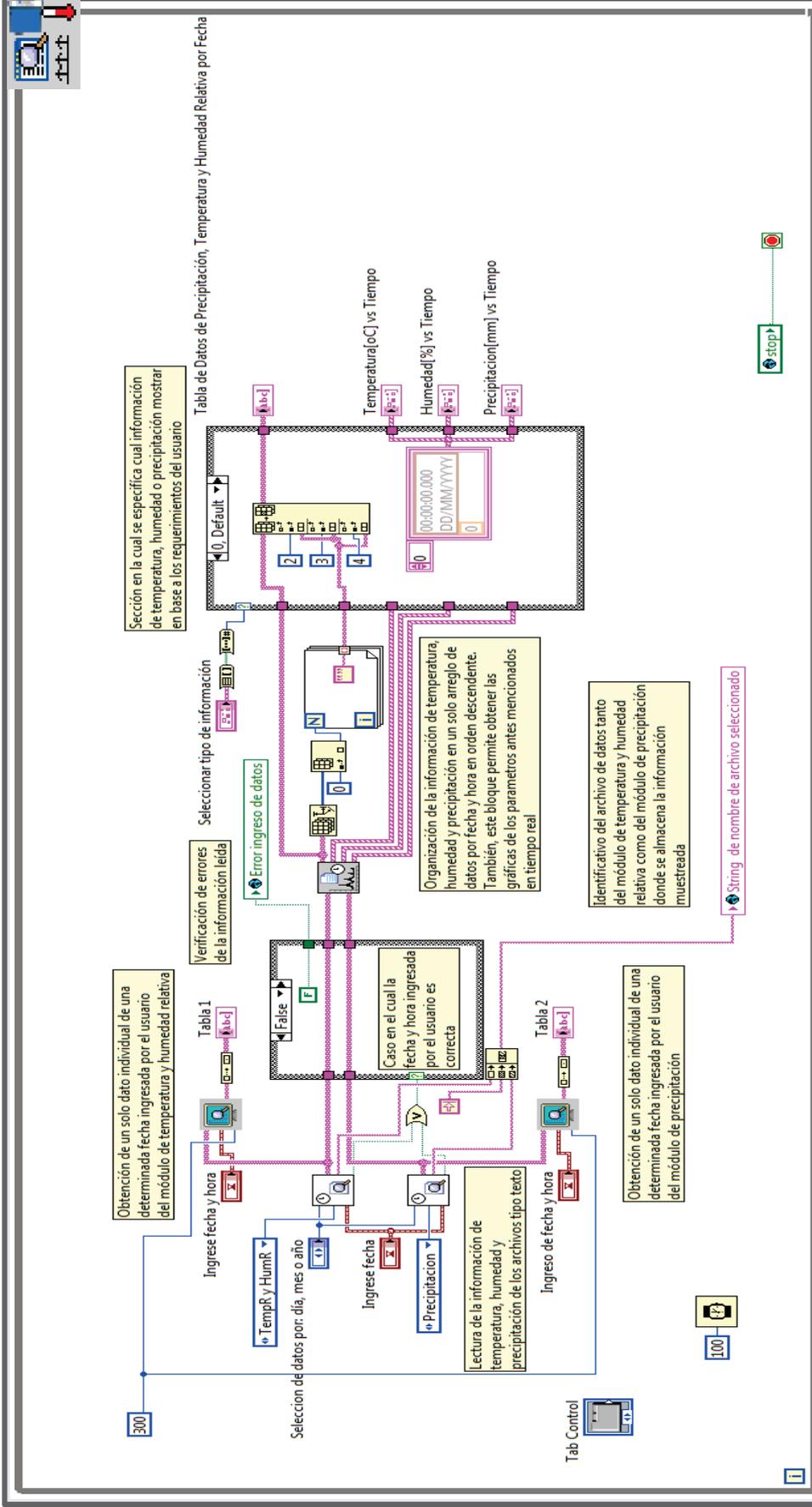
**NOMBRE:** Miguel Bonilla

**TIPO:** Vi



Función que permite organizar la información recibida, en dos archivos de texto que almacenarán la información de temperatura y humedad organizada por fecha y hora y la información de precipitación organizada por fecha y hora respectivamente. Los archivos de texto se crearán cada día, y la información obtenida será muestreada cada 5 minutos.

<b>TÍTULO:</b>	<b>Diagramas de Bloques Inicial de la Recopilación de Datos GPRS</b>	<b>MÓDULO:</b>	<b>Recopilación de Datos GPRS</b>
<b>ANEXO:</b>	<b>C7</b>	<b>NOMBRE:</b>	<b>Miguel Bonilla</b>
	<b>1 de 1</b>	<b>TIPO:</b>	<b>VI</b>



**TÍTULO:** Diagrama de Bloques Inicial de la Presentación de Datos GPRS

**MÓDULO:** Presentación de Datos GPRS

**ANEXO:** C8

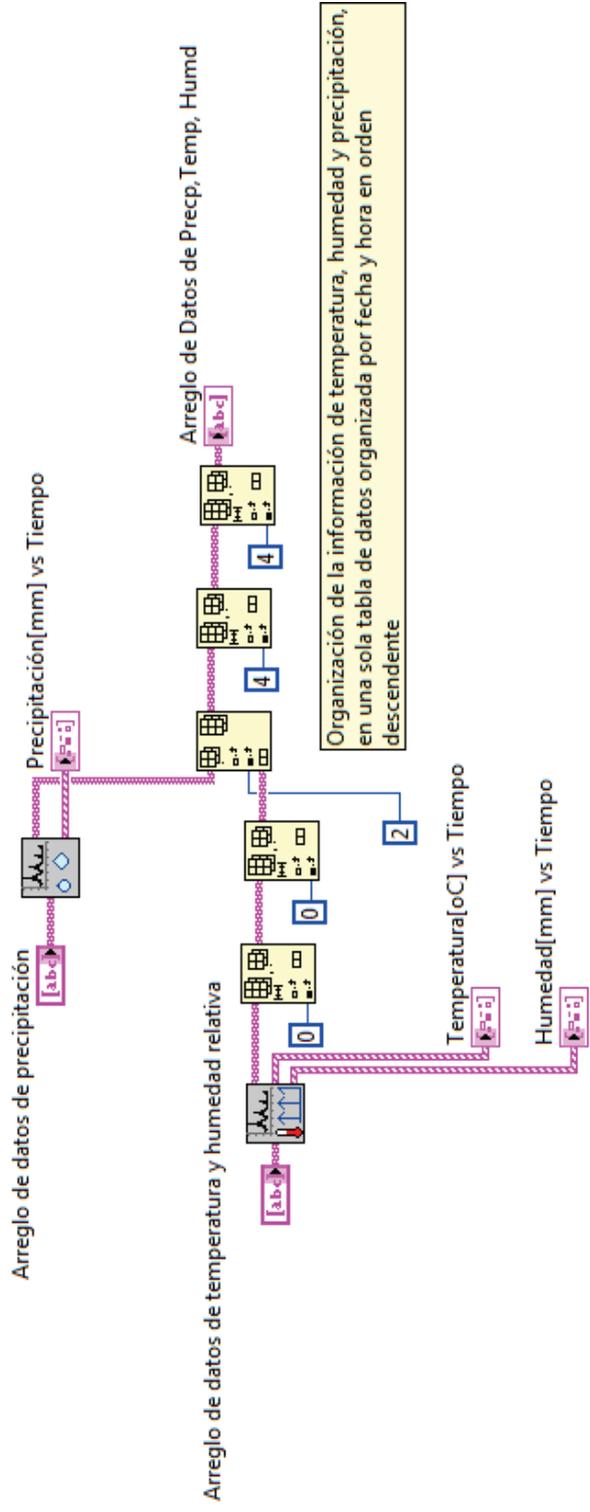
**ESQUEMA No:** 1 de 2

**NOMBRE:** Miguel Bonilla

**TIPO:** Vi



Obtención de la gráfica de precipitación vs tiempo y de una tabla de precipitación organizada por fecha y hora



Obtención de las gráficas de temperatura vs tiempo y humedad vs tiempo, así como la tabla de temperatura y humedad organizada por fecha y hora

<b>TÍTULO:</b> Gráficos y Organización de Información GPRS	<b>MÓDULO:</b> Presentación de Datos GPRS
<b>ANEXO:</b> C8.1	<b>ESQUEMA No:</b> 2 de 2
<b>NOMBRE:</b> Miguel Bonilla	<b>TIPO:</b> Subvi