

**DISEÑO Y CONSTRUCCIÓN DE UN DISPOSITIVO PARA LA SUPERVISIÓN DE
LA PRUEBA DE CONSUMO DE SACAROSA EN RATAS**

GRUPO DE INVESTIGACIÓN

Grupo de investigación en Bioingeniería, Señales y Microelectrónica (BISEMIC).

AUTOR

Juan Andrés Durán Sánchez

**UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA DE INGENIERÍA Y ADMINISTRACIÓN
FACULTAD DE INGENIERÍA ELECTRÓNICA
BUCARAMANGA
2008**

**DISEÑO Y CONSTRUCCIÓN DE UN DISPOSITIVO PARA LA SUPERVISIÓN DE
LA PRUEBA DE CONSUMO DE SACAROSA EN RATAS**

Juan Andrés Durán Sánchez

Trabajo de grado para optar al título de
Ingeniero Electrónico.

DIRECTOR

Claudia Leonor Rueda Guzmán
Ingeniera Electrónica Msc. Informática

GRUPO DE INVESTIGACIÓN

Grupo de investigación en Bioingeniería, Señales y Microelectrónica (BISEMIC).

**UNIVERSIDAD PONTIFICIA BOLIVARIANA
ESCUELA DE INGENIERÍA Y ADMINISTRACIÓN
FACULTAD DE INGENIERÍA ELECTRÓNICA
BUCARAMANGA
2008**

TABLA DE CONTENIDO

	pág.
1. INTRODUCCIÓN.....	1
1.1 MARCO TEÓRICO Y ESTADO DEL ARTE	2
1.2 TRASTORNOS COMPORTAMENTALES	3
1.3 MODELOS ANIMALES.....	4
1.4 ALGUNOS DISPOSITIVOS PARA LA SUPERVISIÓN DE LA PRUEBA DE CONSUMO DE SACAROSA EN RATAS	5
1.4.1 TSE LabMaster – Modular Animal Monitoring System	5
1.4.2 TSE Drinking & Feeding Monitoring System.....	6
1.4.3 Oxymax Lab Animal Monitoring System: CLAMS de Columbus instruments	8
1.5 DISPOSITIVOS BLUETOOTH.....	9
1.5.1 Generalidades de la tecnología Bluetooth	9
1.5.2 Descripción de la tecnología.....	10
1.5.3 La radio Bluetooth	11
1.5.4 Interconexión entre sistemas abiertos (OSI) y el estándar Bluetooth IEEE 802.15.1.....	12
1.5.5 Pila de protocolos Bluetooth.....	13
1.5.5.1 Descripción de los protocolos	15
1.5.5.1.1 LMP	15
1.5.5.1.2 HCI	15
1.5.5.1.3 L2CAP	15
1.5.5.1.4 RFCOMM	15
1.5.5.1.5 SDP	16
1.5.6 Topología de conexión de las WPAN Bluetooth.	16
1.5.6.1 Picoredes WPAN Bluetooth.	16
1.5.6.2 Redes dispersas WPAN Bluetooth.	17
1.6 ESTRUCTURA DEL PROYECTO	18
1.6.1 Planteamiento.....	18
1.6.2 Objetivo General.....	18
1.6.3 Objetivos Específicos	18
2. MÓDULOS DEL DISPOSITIVO	19
2.1 DIAGRAMA DE BLOQUES	19
2.2 MÓDULO TECLADO MATRICIAL 4X4.....	21
2.2.1 Manejo del teclado matricial 4X4.....	22
2.3 MÓDULO LCD 2X16	29

2.3.1	Manejo del LCD 2X16	31
2.4	MÓDULO MEMORIA EEPROM 24LC512.....	36
2.4.1	Descripción de los pines.....	37
2.4.2	Datos a almacenar en memoria.....	38
2.4.3	Manejo de la memoria 24LC512.....	39
2.4.3.1	Escritura de la memoria 24LC512.....	39
2.4.3.2	Lectura de la memoria 24LC512.....	41
2.5	MÓDULO RTC DS1302.....	47
2.5.1	Descripción de los pines.....	47
2.5.2	Manejo del RTC DS1302.....	49
2.5.3	Lectura de la fecha y la hora	52
2.6	MÓDULO DE COMUNICACIÓN BLUETOOTH.....	56
2.6.1	Descripción de los pines de módulo USART y del modem Bluetooth - BlueSMiRF RP- SMA.....	56
2.6.2	Módulo USART del PIC 16F877a.....	57
2.6.2.1	Registros del módulo USART de PIC 16F877a	58
2.6.2.2	Manejo del módulo USART del PIC 16F877a.....	61
2.7	MÓDULO MULTIPLEXORES ANALÓGICOS	61
2.7.1	Descripción de los pines del CD4051	62
2.7.2	Manejo de los multiplexores CD4051	64
2.8	MODULO A/D.....	65
2.8.1	Registros módulo A/D del PIC 16F877a.....	66
2.8.2	Manejo del módulo A/D	68
3.	labview 7.1.....	72
3.1	INSTRUMENTOS VIRTUALES (VI'S) EN LABVIEW	72
3.1.1	Sub VI'S	72
3.2	Comunicación entre el modem Bluetooth – BlueSMiRF RP- SMA y la aplicación de LabVIEW.	73
3.2.1	Funciones para la lectura y escritura de datos en el PC	73
3.2.1.1	Función de lectura de datos de LabVIEW para el PC.....	74
3.2.1.2	Función de escritura de datos de LabVIEW para el PC.....	75
3.2.2	Funciones para la lectura y la escritura de datos en la PocketPC....	76
3.2.2.1	Función de lectura de datos del LabVIEW para la PocketPC	77
3.2.2.2	Función de escritura de datos de LabVIEW para la PocketPC.....	77
4.	FUNCIONAMIENTO.....	79
4.1	MODO DIRECTO	79
4.1.1	Supervisión en el modo directo.	80
4.2	MODO PC	85
4.2.1	Supervisión en modo PC.....	86

4.2.2	Lectura de datos en modo PC	90
4.3	MODO PDA	94
4.3.1	Supervisión en modo PDA.....	94
5.	CONCLUSIONES	100
6.	BIBLIOGRAFÍA.....	101

LISTA DE FIGURAS

	Pág.
Figura 1. TSE LabMaster – Modular Animal Monitoring System.....	5
Figura 2. TSE Drinking & Feeding Monitoring System.....	7
Figura 3. interfaz gráfica del TSE Drinking & Feeding Monitoring System.....	7
Figura 4. Sistema Oxymax Lab Animal Monitoring	8
Figura 5. Esquema de interconexión de diversos dispositivos Bluetooth.....	9
Figura 6. Símbolo de la tecnología Bluetooth	10
Figura 7. Capas del protocolo Bluetooth en el modelo OSI	13
Figura 8. Distintos bloques funcionales de un sistema Bluetooth	14
Figura 9. Pila de protocolos	14
Figura 10. Topología de una red dispersa Bluetooth WPAN.....	17
Figura 11. Diagrama de bloques del proyecto	19
Figura 12. Esquema circuital de un teclado matricial 4X4.....	21
Figura 13. Diagrama circuital del teclado matricial 4X4	22
Figura 14. Diagrama de flujo de la rutina de manejo del teclado matricial 4X4.....	23
Figura 15. Fotografía de un LCD 2X16	30
Figura 16. Conexión del LCD con bus de 8 bits.....	30
Figura 17. Conexión del LCD con bus de 4 bits.....	30
Figura 18. Diagrama circuital de LCD 2X16.....	31
Figura 19. Diagrama circuital de la memoria EEPROM 24LC512.....	37

Figura 20. Diagrama de flujo para la lectura de los datos de la memoria 24LC512	42
Figura 21. Diagrama circuital del RTC DS1302	48
Figura 22. Diagrama de flujo de la lectura de la fecha y la hora	53
Figura 23. Diagrama circuital del modulo Bluetooth.....	56
Figura 24. Estructura del registro TXSTA	58
Figura 25. Estructura del registro RCSTA.....	59
Figura 26. Diagrama circuital de los multiplexores analógicos CD4051.....	62
Figura 27. Diagrama de flujo de la rutina de manejo de los CD4051.....	64
Figura 28. Estructura del registro ADCON0.....	66
Figura 29. Estructura del registro ADCON1	67
Figura 30. Diagrama de flujo de las adquisición de señales del dispositivo de supervisión.....	69
Figura 31. Función de configuración de la comunicación	73
Figura 32. Función de lectura de datos de LabVIEW para el PC.....	74
Figura 33. Función de escritura de datos de LabVIEW para el PC.....	75
Figura 34. Función de configuración del puerto.....	76
Figura 35. Función de lectura en LabVIEW usada para la PocketPC.....	77
Figura 36. Función de escritura de LabVIEW usada para la PocketPC.....	78
Figura 37. Mensaje de bienvenida para el usuario	79
Figura 38. Mensaje del menú principal del dispositivo.....	79
Figura 39. Mensaje de confirmación del ingreso de los datos	80

Figura 40. Mensaje para elegir el porcentaje de sacarosa que se utilizará en la prueba.....	80
Figura 41. Mensaje de confirmación de porcentaje de concentración	81
Figura 42. Mensaje de elección de tiempos de duración	81
Figura 43. Mensaje de confirmación del tiempo de duración.....	82
Figura 44. Mensaje para el ingreso del número de ratas	82
Figura 45. Mensaje de error de números	82
Figura 46. Mensaje de error de ingreso de letras	83
Figura 47. Mensaje de confirmación del número de ratas	83
Figura 48. Mensaje de indicaciones para el ingreso de los ID de las ratas	83
Figura 49. Mensaje de ingreso del ID de la rata	84
Figura 50. Confirmación del ID de la rata ingresado.....	84
Figura 51. Mensaje de adquisición en curso.....	84
Figura 52. Mensaje de fin de la prueba.....	85
Figura 53. Menú del modo PC	85
Figura 54. Interfaz gráfica de usuario del modo PC	86
Figura 55. Mensaje de confirmación de la supervisión a través del PC	87
Figura 56. Mensaje de invitación a iniciar la prueba desde el PC	87
Figura 57. Interfaz de control de la supervisión a través del PC	88
Figura 58. Diagrama de transición de estados del modo supervisar con PC.....	89
Figura 59. Tabla de resultados obtenidos de la aplicación de una prueba	90
Figura 60. Mensaje de confirmación para la lectura de los datos a través del PC.....	91
Figura 61. Mensaje de invitación a iniciar lectura desde el PC	91

Figura 62. Interfaz de control del modo lectura de datos con el PC.....	92
Figura 63. Diagrama de transición de estados del modo lectura con PC	93
Figura 64. Tabla de resultado de la lectura de los datos almacenado en el dispositivo	94
Figura 65. Mensaje de confirmación de supervisión de la prueba en modo PDA ..	95
Figura 66. Mensaje de invitación a iniciar la prueba desde la PocketPC	95
Figura 67. Interfaz gráfica de usuario del modo PDA	96
Figura 68. Interfaz de control del modo PDA	97
Figura 69. Interfaz de de detección del puerto Bluetooth.....	98
Figura 70. Diagrama de transición de estados del modo supervisar con la PocketPC	99

LISTA DE TABLAS

	pág.
Tabla 1. Rutina en C para el manejo del teclado matricial 4X4.....	24
Tabla 2. Rutina lcd.c, tomada de los drivers de CSS C compliler	32
Tabla 3. Datos almacenados por cada rata	38
Tabla 4. Rutina 24512.c, tomada de los drivers de CSS C Compiler.....	40
Tabla 5. Rutina en C para la lectura de la memoria 24LC512	43
Tabla 6. Rutina DS1302.c, tomada de los drivers de CSS C Compiler.....	49
Tabla 7. Rutina en C para la lectura de la fecha y hora del DS1302	54
Tabla 8. Rutina en c para conversión de los datos HEX a ASCII o decimal	54
Tabla 9. Configuración del USART en C.....	61
Tabla 10. Tabla de verdad del CD 4051	63
Tabla 11. Rutina en C para el control de los multiplexores CD4051	65
Tabla 12. Posibles configuraciones de los canales del módulo A/D del PIC 16F877a.....	68
Tabla 13. Código en C de la adquisición A/D con el PIC16F877a.....	70

RESUMEN

En este documento, se muestra el diseño y construcción de un dispositivo electrónico para la supervisión de la prueba de consumo de sacarosa en ratas. El objetivo de este dispositivo, es la de supervisar la prueba de consumo de sacarosa en ratas mediante tres modos de manejo nombrados así: modo directo, modo PC y modo PDA; cada uno de estos modos de manejo cuenta con una interfaz gráfica de usuario en el dispositivo y para el caso de los modos PC y PDA posee interfaces de usuario desarrolladas en LabVIEW 7.1 para la cada uno de ellas. La comunicación entre el dispositivo y el PC o PDA se realiza mediante el uso de la especificación Bluetooth a través del Modem Bluetooth - BlueSMiRF RP-SMA, el control del dispositivo es realizada por un PIC 16F877a.

Este documento está dividido en 5 capítulos en los que se muestra la etapa investigativa y la etapa de diseño y construcción del dispositivo de supervisión.

En el primer capítulo se muestra el estado del arte de y el marco teórico necesario para el diseño. En el segundo capítulo se exponen los módulos de hardware que constituyen el dispositivo, así como los algoritmos que se implementaron para la construcción del dispositivo. En el tercer capítulo se muestra la forma en la que se establece la comunicación entre el dispositivo de supervisión y el PC o la PocketPC usando una aplicación de LabVIEW 7.1. En el cuarto capítulo se explican los tres modos de manejo del dispositivo de supervisión. En el quinto y último capítulo se dan las conclusiones arrojadas después de la investigación que llevo a la construcción del dispositivo de supervisión.

ABSTRACT

In this document shows the design and construction of an electronic device for monitoring the test sucrose consumption in rats. The purpose of this device, is to oversee the trial of sucrose consumption in rats through three modes of management appointed as follows: direct mode, so PC and PDA mode, each of these modes of management with a graphical user interface in the device and methods for PC and PDA has user interfaces developed in LabVIEW 7.1 for each of them. Communication between the device and the PC or PDA is done using the Bluetooth specification through Bluetooth Modem - BlueSMiRF RP-SMA, controlling device is made by a PIC16F877a. This document is divided into 5 chapters that shows the investigative stage and the stage design and construction supervision of the device. The first chapter shows the state of the art and the theoretical framework necessary for the design. The second chapter outlines the hardware modules that make up the device and algorithms that were implemented for the construction of the device. In the third chapter shows the way in which established communication between the monitoring device and the PC or PocketPC using an application LabVIEW 7.1. In the fourth chapter explains the three modes of operation monitoring device. In the fifth and final chapter gives the conclusions dropped after the investigation that mild to the construction of the monitoring device.

1. INTRODUCCIÓN

Con el propósito de estudiar manifestaciones comportamentales del tipo de la ansiedad y la depresión en seres humanos, se han diseñado una serie de pruebas psicológicas con animales especialmente ratas para medir estas manifestaciones, una de estas pruebas es la que evalúa el consumo de sacarosa en una rata que es sometida a un aislamiento social crítico.

Dado que la prueba de consumo de sacarosa en ratas es una técnica de evaluación de naturaleza experimental se requiere del uso de una muestra considerable de individuos para poder entregar conclusiones reales, lo que implica tomar gran cantidad de datos y en muchos casos inversión de tiempo con el riesgo a que la prueba fracase y solo se conoce de estas situaciones cuando esta se termina.

En este proyecto se buscó diseñar y construir un dispositivo capaz de supervisar la prueba de consumo de sacarosa en ratas, la prueba consiste en someter a una rata a un aislamiento social crónico en una jaula y colocar agua y una solución de sacarosa en recipientes independientes para que la rata tome de los líquidos durante un tiempo determinado y luego se evalúa el consumo de ambos líquidos para sacar los análisis y conclusiones pertinentes. El dispositivo construido supervisa de forma automática la prueba de consumo de sacarosa en ratas así como de forma remota, utilizando un PC (personal computer) o una PDA (personal digital assistant), la importancia de que la supervisión se realice de forma automática y remota influye en que se cumpla la condición de aislamiento social crítico y además se pueda controlar de forma automática el registro de datos.

Este proyecto se desarrolló dentro del grupo de investigación en Bioingeniería, Señales y Microelectrónica (BISEMIC) y tiene con fin ofrecer al grupo de neurociencias de la Universidad Pontificia Bolivariana, un dispositivo que les permita la supervisión de la prueba de consumo de sacarosa en ratas.

Este documento está dividido en 5 capítulos en los que se muestra la etapa investigativa y la etapa de diseño y construcción del dispositivo de supervisión.

En el primer capítulo se muestra el estado del arte de y el marco teórico necesario para el diseño. En el segundo capítulo se exponen los módulos de hardware que constituyen el dispositivo, así como los algoritmos que se implementaron para la construcción del dispositivo. En el tercer capítulo se muestra la forma en la que se establece la comunicación entre el dispositivo de supervisión y el PC o la PocketPC usando una aplicación de LabVIEW 7.1. En el cuarto capítulo se explican los tres modos de manejo del dispositivo de supervisión. En el quinto y último capítulo se dan las conclusiones arrojadas después de la investigación que llevo a la construcción del dispositivo de supervisión.

1.1 MARCO TEÓRICO Y ESTADO DEL ARTE

La parte inicial del proyecto consistió en la recopilación de información referente a los estudios de las manifestaciones comportamentales de tipo depresión – ansiedad en ratas sometidas al aislamiento social crónico utilizando como medio de estimulación la prueba de consumo de sacarosa en ratas, en proyectos de investigación desarrollados en el país o en otros países, para de esta manera tener una idea más clara que ayudara al diseño final del dispositivo.

En Santander y el resto de Colombia no se conocen desarrollo de dispositivos para supervisar la prueba de consumo de sacarosa en ratas y se han realizado pocos estudios sobre la evaluación de las manifestaciones comportamentales de tipo depresión – ansiedad en ratas sometidas al aislamiento social crónico utilizando como medio de estimulación la prueba de consumo de sacarosa en ratas. El grupo de investigaciones de neurociencias y comportamiento de la Universidad Industrial de Santander – Universidad Pontificia Bolivariana han desarrollado investigaciones sobre este tema específico.

En la parte que hace referencia a los dispositivos de supervisión para la prueba de consumo de sacarosa en ratas no existen registros en la región de que se hayan realizado desarrollos orientados a hacia este tema.

En Estados Unidos existen empresas que ofrecen sistemas completos para supervisar el consumo de sacarosa en ratas e incluso de alimento, una de las más importantes es TSE Systems, la cual en su portafolio de productos nos presenta:

- El TSE LabMaster – Modular Animal Monitoring System que es un sistema avanzado que supervisa en ratas aisladas parámetros como: consumo de líquidos y sólidos, el desempeño del metabolismo, la actividad en la jaula, para 24 jaulas simultáneas usando un PC.¹
- El TSE Drinking & Feeding Monitoring System, el cual es un sistema avanzado utilizado para la grabación y el análisis de consumo de alimento sólido y líquido de animales de laboratorio entre ellos ratas.²

Por otro lado la compañía Columbus Instruments de Estados Unidos también ha desarrollado un sistema de supervisión para animales pequeños de laboratorio que se describe a continuación:

¹ <http://tse-systems.com/labmaster/labmaster.htm>

² <http://tse-systems.com/drinking-feeding/drinking-feeding.htm>

- El Oxymax Lab Animal Monitoring System: CLAMS, es un sistema de control que reúne una amplia gama de mediciones y las integra para la supervisión de un número máximo de 32 ratas u otro animal pequeño de laboratorio.³

1.2 TRASTORNOS COMPORTAMENTALES

En el mundo existe una preocupación por los desordenes mentales especialmente el de la depresión mayor (MDD). En el contexto que se refiere a los países en desarrollo, esta enfermedad constituye el cuarto problema de salud pública más importante, con una tasa de mortalidad de 3.4 %. No cabe duda que estos padecimientos generan condiciones desadaptativas que son una causa importante de incapacidad laboral, sobrellevando grandes pérdidas económicas.

Los desordenes de depresión mayor están altamente relacionados con la inhabilidad de realizar actividades normales, problemas en el funcionamiento social y cognoscitivo. Se caracteriza por una alteración principal en el afecto, con episodios que pueden durar varias semanas o meses; los pacientes sufren incapacidad de experimentar placer y pérdida de interés, con sentimientos de infravaloración, culpa y déficit cognitivo.⁴

Los trastornos de ansiedad también se constituyen en uno de los desordenes mentales consecuentes a la exposición del individuo a diferentes factores adversos. La ansiedad se define como un estado emocional subjetivamente vivenciado, como desagradable y amenazador, que produce alteraciones de humor y cognición generalmente acompañados por alteraciones comportamentales y fisiológicas.⁵ En cualquier momento de nuestras vidas hemos experimentado ansiedad, un sentimiento de aprensión o de tensión ante situaciones de estrés. Reacción normal y necesaria para la supervivencia de los individuos y de nuestra especie, esta forma de ansiedad no representa disfuncionalidad, ya que en alguna medida, la ansiedad es una reacción ante el estrés que ayuda en el funcionamiento diario de las personas. Sin ansiedad, los seres humanos careceríamos de la motivación necesaria para vivir y desarrollar las actividades diarias.⁶

³ <http://www.colinst.com/brief.php?id=61>

⁴ Páez X, L. Hernández, T. Baptista. Avances en la terapéutica molecular de la depresión: revista de neurología, 2003

⁵ Conde C., Bothelo S., Berena A., Velásquez M. La vía visual puede ser el dissipador de ansiogenicidad en el modelo del laberinto en cruz elevado. – Colombia: Salud UIS, 2001.

⁶ Bekker, H.L., Legare F., Stacey D., O'connor A., Lemyre L. in anxiety a suitable measure of decision aid effectiveness: A systematic review?. Patient education and counseling, 2003.

1.3 MODELOS ANIMALES

No cabe duda que los trastornos de ansiedad y depresión son entidades complejas y de alta prevalencia a nivel mundial y nacional, sin embargo la fisiopatología y los mecanismos farmacológicos asociados, son aun poco conocidos: Así, por razones éticas y metodológicas se requiere el desarrollo y utilización de modelos animales que permitan la investigación controlada de los procesos neurobiológicos a ellas asociados. En este sentido, considerando la importancia de determinar los síntomas de la ansiedad y de la depresión y partiendo de que la sacarosa es un reforzador primario, algunos estudios han utilizado la prueba de consumo de sacarosa en ratas para la comprensión de los procesos neurobiológicos adyacentes a esta patología. Dentro de este contexto, la ansiedad es interpretada operacionalmente como un decrecimiento en la sensibilidad de recompensas: en algunos estudios se ha encontrado que el perfil ansioso de los roedores está asociado con una reducción en el consumo de soluciones de baja concentración de sacarosa. Por lo tanto desde el punto de vista operacional algunos estudios consideran la ansiedad como una disminución en el consumo de sacarosa, mientras que otros la consideran como un aumento en el consumo de sacarosa en soluciones concentradas.⁷

La prueba de consumo de sacarosa va de la mano del modelo de aislamiento social para inducir estrés crónico en ratas, fundamentado en algunos estudios que demostraron la importancia de los efectos de ciertos acontecimientos al inicio de la vida sobre el desarrollo posterior y la aparición de la depresión; en particular, la importancia de las experiencias precoces de pérdida o privación maternal.

Este modelo consiste en tomar la rata del laboratorio y someterla a aislamiento social temprano. Se conduce desde el destete a la edad adulta y priva animales del contacto social durante la fase crítica de la vida que se caracteriza por el juego social; en este caso los animales son alojados individualmente y mantiene contactos olfativos, visuales y auditivos, pero no contacto físico; la carencia de contacto físico genera una gama de reacciones comportamentales y fisiológicas que afectan considerablemente la reactividad emocional de las ratas adultas, produciendo un síndrome comportamental que incluye hiperactividad en un ambiente nuevo, un incremento en la exploración, una respuesta aumentada a los esfuerzos y perseverancia.⁸ Se ha demostrado que la exposición de este tipo de ratas a estímulos estresores aumenta los comportamientos asociados tanto de depresión como de ansiedad, pues la exposición de roedores a la tensión imprevisible ha dado lugar a los perfiles del comportamiento indicados de estados

⁷ Sammuta S., Bethusb I., Goodallb G., Muscata R. antidepressant reversal of interferon- a- induced anhedonia, *Physiology & Behavior.* , 2002

⁸ Botelho S., Conde C., modelos animales para el estudio de ansiedad y depresión. Colombia: salud UIS, 2001.

psicopatológicos humanos, mientras que se proporciona el mayor control de variables experimentales.⁹

1.4 ALGUNOS DISPOSITIVOS PARA LA SUPERVISIÓN DE LA PRUEBA DE CONSUMO DE SACAROSA EN RATAS

Como preámbulo para el diseño del dispositivo para la supervisión de consumo de sacarosa en ratas se estudiaron ciertos sistemas que ofrecen algunas compañías de otros países sobre el particular, con el propósito de realizar el análisis y los estudios necesarios que permitieran efectuar las adecuaciones para el diseño del dispositivo.

1.4.1 TSE LabMaster – Modular Animal Monitoring System

El TSE TSE LabMaster es un sistema modular automatizado de vigilancia de las conductas metabólicas y de comportamiento de los animales pequeños de laboratorio, en la Figura 1 se muestra el sistema.

Figura 1. TSE LabMaster – Modular Animal Monitoring System



Fuente:http://tse systems.com/download/TSE_Metabolism_LabMaster_Phenomaster_2007-09-A_high_resol.pdf

⁹ [25] Matuszewich L., Karney. J.J., Crter S., Janasik S., O'Brien J. Friedman R. the delayed effects of cronic unpredictable stress on anxiety measures. Physiology & Behavior, 2007.

Algunas de las características más relevantes del sistema del sistema de monitoreo de animales pequeños de laboratorio son:¹⁰

- Medidas de rendimiento metabólico a través de calorimetría indirecta - O₂ consumo y la producción de CO₂, tipo de Cambio respiratorio (RER), el gasto energético (EE).
- Mediciones de conducta de alimentación a través de sensores de alta precisión, con la capacidad de manejo de hasta 6 de sensores por jaula, controlada el acceso de los alimentos o líquidos por tiempo o cantidad.
- Jaula con sensor de actividad a través de marcos, que determina la actividad total de los animales.
- Rueda de medición de la actividad atlética de forma voluntario o forzada, medición de la carga de control de trabajo; tiene opciones de control de activación o desactivación de la función a través del PC.
- Medición de la orina y las heces en jaulas metabólicas a través de sensores de cuantificación de peso.
- Jaula para el medio ambiente imparcial sin estrés para el comportamiento del animal.
- El sistema viene con el software LabMaster para establecer el experimento y diseño para el registro y la evaluación de los datos de los ensayos. Los datos medidos se pueden exportar a Excel u otras bases de datos estándar.

1.4.2 TSE Drinking & Feeding Monitoring System

Es un sofisticado sistema diseñado para la supervisión automática y el análisis en el consumo de los alimentos y líquidos de todos los animales pequeños de laboratorio. Este sistema de supervisión permite un número ilimitado de estaciones de medición que puede ser controlada por un solo PC.

¹⁰ <http://tse-systems.com/labmaster/labmaster.htm>

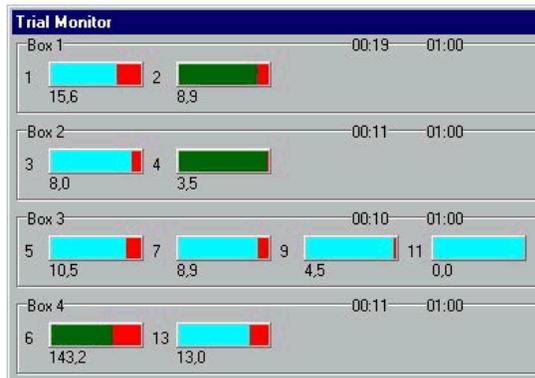
Figura 2. TSE Drinking & Feeding Monitoring System



Fuente: <http://tse-systems.com/drinking-feeding/drinking-feeding.htm>

En cuanto a la parte de adquisición de datos Varias jaulas de prueba se pueden iniciar al mismo tiempo si es necesario. Durante la adquisición, las señales provenientes de los sensores son supervisadas cada segundo. Para compensar las vibraciones del sistema lleva a cabo un control de dispersión: si la señal de variación es inferior a un límite preestablecido los datos se almacenan de lo contrario son rechazados. Cuenta con una interfaz gráfica de usuario que permite observar la situación actual de llenado de las jaulas, de tal manera que el operador pueda ver los nivel de en cualquier momento¹¹. En la figura 3 se muestra la interfaz gráfica de usuario con la que cuenta el sistema.

Figura 3. interfaz gráfica del TSE Drinking & Feeding Monitoring System



Fuente: <http://tse-systems.com/drinking-feeding/drinking-feeding.htm>

¹¹ <http://tse-systems.com/drinking-feeding/drinking-feeding.htm>

1.4.3 Oxymax Lab Animal Monitoring System: CLAMS de Columbus instruments

Es un laboratorio con un sistema de control que reúne una amplia gama de mediciones y las integra en un sistema capaz de la supervisión de 32 ratas al mismo tiempo. El Oxymax sirve como una plataforma en donde los usuarios pueden supervisar la actividad, alimentación y bebida de las ratas, gracias a un sistema llamado IC – Bus que permite la fácil integración de los datos. Dentro de cada jaula se puede elegir de forma no invasiva la recolección simultánea de parámetros de alimentación de bebida y de rendimiento metabólico, de actividad y de temperatura¹². En la Figura 4 se puede apreciar el sistema Oxymax funcionando

Figura 4. Sistema Oxymax Lab Animal Monitoring



Fuente: <http://www.colinst.com/brief.php?id=61>

¹² <http://www.colinst.com/brief.php?id=61>

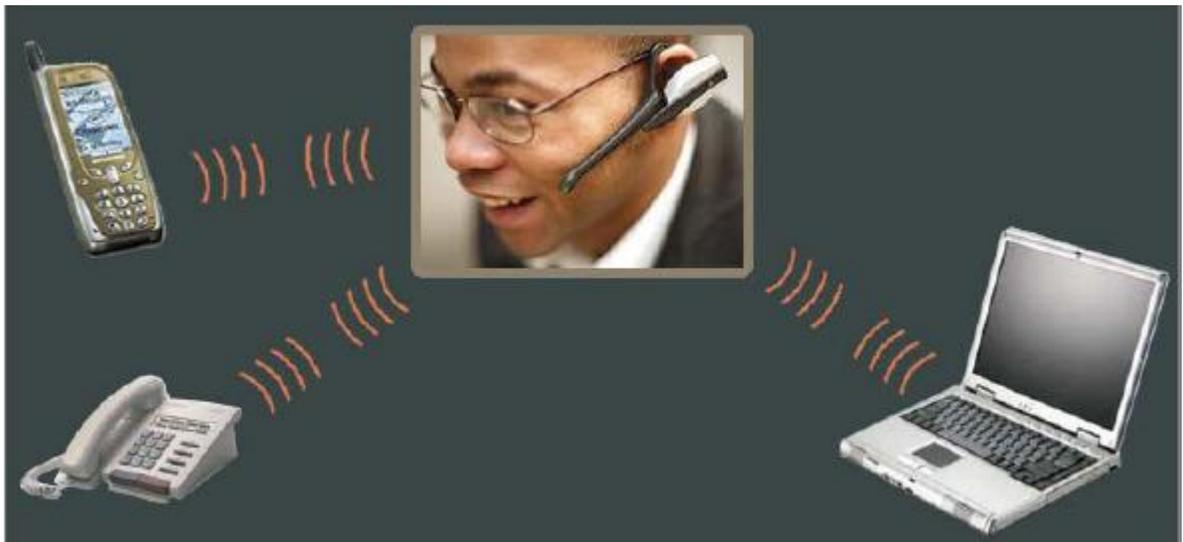
1.5 DISPOSITIVOS BLUETOOTH

1.5.1 Generalidades de la tecnología Bluetooth

El concepto de Bluetooth surge de la imperiosa necesidad de las grandes empresas de telecomunicaciones e informática, de desarrollar una interfaz abierta, que facilite la comunicación entre los diferentes equipos informáticos y telefónicos, aprovechando la capacidad y la movilidad de los dispositivos inalámbricos, para la total supresión de los cables de conexión, y adoptar así un único estándar de comunicación.

Una de las principales ventajas de la tecnología inalámbrica Bluetooth es su capacidad para gestionar simultáneamente tanto transmisiones de voz como de datos. Esto permite a los usuarios disfrutar de una gran variedad de soluciones innovadoras, tales como el uso de manos libres para atender llamadas, funciones de impresión y fax, o la sincronización de aplicaciones entre PDA, PC y móviles, entre otras muchas como se muestra en la figura 15.¹³

Figura 5. Esquema de interconexión de diversos dispositivos Bluetooth



Fuente: <http://campusvirtual.uma.es/siselmed/Transparencias07/bluetooth.pdf>

El Bluetooth Special Interest Group (SIG) o en español grupo de interés especial, es una asociación comercial formada por líderes en telecomunicación, informática,

¹³ <http://spanish.bluetooth.com/Bluetooth/Technology/Basics.htm>

automatización industrial, industrias de tecnología en redes de datos, encargada del desarrollo e introducción en el mercado de la tecnología inalámbrica Bluetooth. Entre algunas de las compañías que han promovido el desarrollo de la tecnología Bluetooth en el mundo tenemos, Agere, Ericsson, IBM, Intel, Microsoft, Motorola, Nokia y Toshiba, centenares de compañías asociadas y otras tantas a nivel mundial que han adoptado el estándar.¹⁴

La oficina central del SIG de Bluetooth está ubicada en Bellevue, Washington, y también dispone de dependencias en Hong Kong y en Malmo, Suecia.

Figura 6. Símbolo de la tecnología Bluetooth



Fuente: www.miblackberry.cl/categoria/consejos/page/2

1.5.2 Descripción de la tecnología

La tecnología Bluetooth opera en una banda de frecuencia industrial, científica y médica (ISM) que no requiere licencia y que se encuadra, concretamente, entre 2.4 y 2.485 GHz. Utiliza una señal bidireccional en un espectro ensanchado por salto de frecuencia a una velocidad nominal de 1600 saltos/segundo. Los saltos de frecuencia se dan entre un total de 79 frecuencias con intervalos de 1Mhz; esto permite dar seguridad y robustez. La banda ISM de 2.4 GHz está disponible en casi todos los países y no suele requerir licencia.

¹⁴ <http://foro.elhacker.net/index.php/topic,83967.0.html>

Básicamente el hardware que compone el dispositivo Bluetooth está compuesto por dos partes. Un dispositivo de radio, encargado de modular y transmitir la señal; y un controlador digital.

A diferencia de otros estándares inalámbricos, la especificación Bluetooth otorga a las empresas de desarrollo definiciones para la capa de enlace y de aplicaciones, lo que permite que sea compatible con soluciones de voz y datos.¹⁵

Se tiene dos versiones de la especificación principal

- La versión 2.0 EDR (transferencia de datos mejorada), adoptada en noviembre de 2004.
- La versión 1.3 adoptada en noviembre de 2003.

Por otra parte se tienen tres clases de dispositivos según la potencia de los mismos:

- Clase 1: son dispositivos de largo alcance (aprox. 100m) utilizados principalmente en la industria, con una potencia máxima de salida de 20dBm.
- Clase 2. Dispositivos de medio alcance (aprox. 10m) utilizados en la gran mayoría de dispositivos móviles actuales, su potencia máxima de salida es de 4dBm.
- Clase 3. Dispositivos de corto alcance (aprox. 10cm), su potencia máxima de salida es de 0dBm.

Las radios más utilizadas son las clase 2, con un consumo de potencia de 2.5mW. Por este motivo la tecnología ha evolucionado para reducir el consumo de potencia, la especificación Bluetooth cambia las radios al modo de ahorro de energía cuando no se encuentran activas.

1.5.3 La radio Bluetooth

Como se ha venido mencionado los dispositivos Bluetooth trabajan en la banda de frecuencia de 2.4 GHz Estos dispositivos usan un transmisor de salto de frecuencia para evitar las interferencias y las pérdidas de intensidad.

¹⁵ <http://spanish.bluetooth.com/Bluetooth/Technology/Basics.htm>

La modulación en una radio Bluetooth se puede realizar en dos modos: uno es el modo obligatorio o de transferencia básica y el otro es el modo opcional o de transferencia mejorada de datos. En el modo de transferencia básica se usa una modulación FSK (Frequency shift keying) la cual consiste en usar como señal moduladora un flujo de pulsos binarios que cambian entre valores determinados. Por su parte el modo de transferencia de mejorada de datos usa una modulación PSK (Phase Shift Keying) que básicamente funciona haciendo variar la fase de la señal portadora entre un número de valores discretos, a su vez la modulación PSK tiene dos variantes la $\pi/4$ -DQPSK y 8DPSK.

La máxima velocidad de transferencia de datos es de 1Mbps para el caso del modo obligatorio o de modulación FSK, 2 Mbps cuando se usa una modulación $\pi/4$ -DQPSK y de 3 Mbps cuando la radio Bluetooth trabaja con una modulación 8DPSK

Las bandas de frecuencia ISM sobre la que trabaja la especificación Bluetooth se organiza en 79 canales que abarquen de 2400 MHz a 2483.5 MHz, entre cada canal se deja una franja de frecuencia de 1 MHz, para evitar saltar las regulaciones que cada país tiene sobre el espectro se utilizan dos bandas de guarda una inferior de 2 MHz y otra superior de 3.5 MHz.

1.5.4 Interconexión entre sistemas abiertos (OSI) y el estándar Bluetooth IEEE 802.15.1

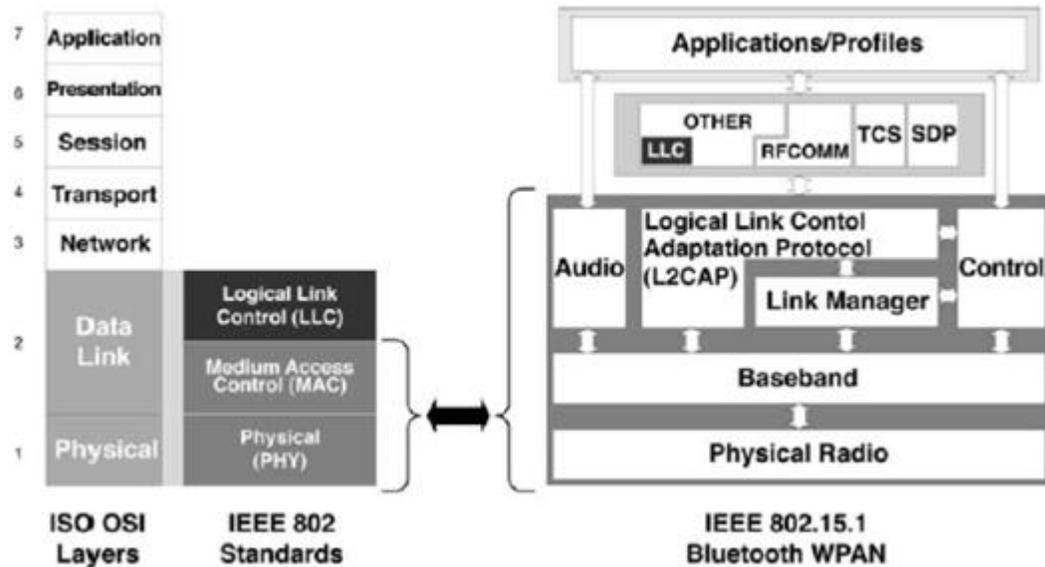
Hay dos formas de ver cualquier diseño de sistemas de comunicación; por su arquitectura y su funcionalidad. Un acercamiento hacia la arquitectura enfatiza las divisiones lógicas de un sistema y como son compatibles. El acercamiento hacia la funcionalidad hace énfasis en los componentes actuales y sus interconexiones.

A continuación se presenta una vista de la arquitectura de una WPAN (ver Figura 17). Se enfatiza la tradicional separación de larga escala del sistema en dos partes; physical layer (PHY) del IEEE 802.15.1 y la subcapa MAC (medium access control) del DLL (data link layer).

En la figura 7 se muestra además la pila del protocolo en la séptima capa del modelo del OSI en la tecnología inalámbrica de Bluetooth y su relación con este estándar. También se puede observar en la figura 7 que las subcapas LLC (logical link control) y el MAC juntas abarcan las funciones deseadas para el DLL del modelo de OSI.¹⁶

¹⁶ http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/archundia_p_fm/capitulo3.pdf

Figura 7. Capas del protocolo Bluetooth en el modelo OSI



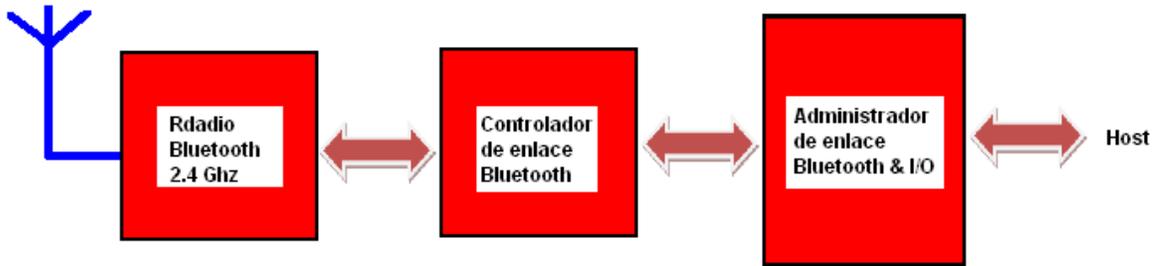
Fuente: http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/archundia_p_fm/capitulo3.pdf

1.5.5 Pila de protocolos Bluetooth¹⁷

Para comunicarse con otros dispositivos Bluetooth, se requiere un hardware específico para Bluetooth, que incluye un módulo de banda base, así como otro módulo de radio y una antena. Además deberá haber un software encargado de controlar la conexión entre dos dispositivos Bluetooth; este software (Link Manager) por lo general correrá en un microprocesador dedicado. Los Link Managers de diferentes dispositivos Bluetooth se comunicarán mediante el protocolo LMP (Link Manager Protocol), como se observa en la figura 8.

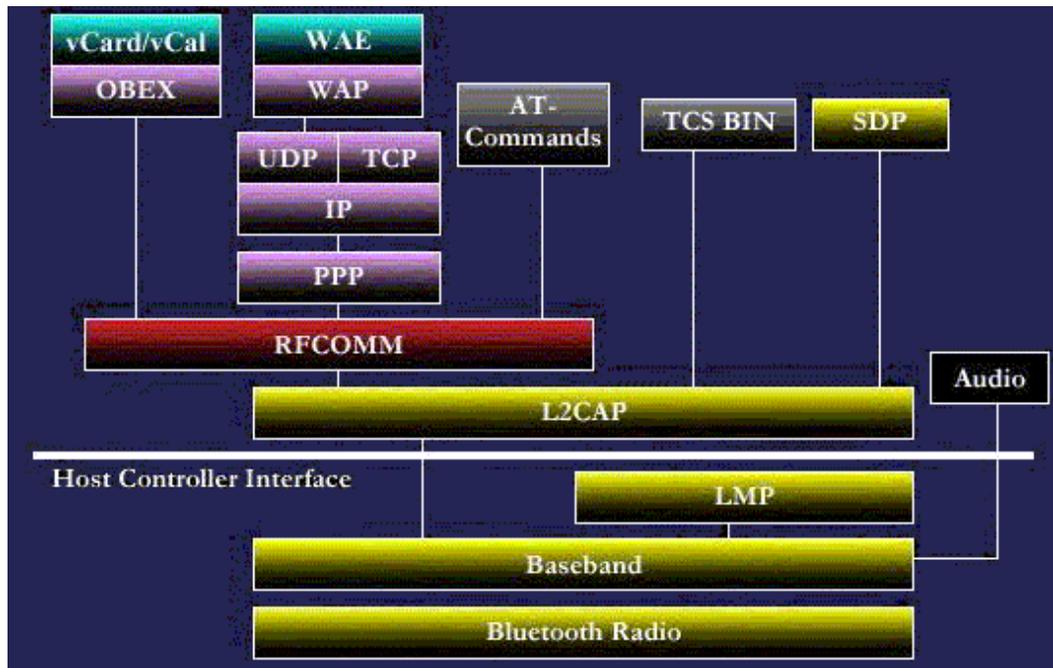
¹⁷ <http://foro.elhacker.net/index.php/topic,83967.0.html>

Figura 8. Distintos bloques funcionales de un sistema Bluetooth



La pila completa se compone tanto de protocolos específicos de Bluetooth (LM (Link Manager) y L2CAP (Logical Link Control Adaption Protocol), por ejemplo como de protocolos no específicos de Bluetooth como son OBEX (Objects Exchange Protocol), UDP (User Datagram Protocol), TCP, IP, etc (ver figura 9). Debido a que la hora de diseñar la torre de protocolos, el objetivo principal ha sido maximizar el número de protocolos existentes que se puedan reutilizar en las capas más altas para diferentes propósitos.

Figura 9. Pila de protocolos



Fuente: <http://usuarios.lycos.es/XESC2000/Projectes/2494/tecno.html#3-4-3>

1.5.5.1 Descripción de los protocolos¹⁸

1.5.5.1.1 LMP

LMP (Link Manager Protocol) controla activamente el establecimiento de comunicación, autenticación y configuración del enlace. Descubre otros gestores de enlace (LM) y se comunica con ellos mediante LMP.

En esencia, este protocolo consiste en un número de PDUs que son enviados desde un dispositivo a otro, determinado por AM_ADDR de la cabecera del paquete. Estos PDUs son siempre enviados como paquetes en un slot individual, y la cabecera del payload es de un byte.

1.5.5.1.2 HCI

La capa HCI provee una interfaz de comandos al controlador de banda base y al gestor del enlace, además de acceder al estado del hardware y a los registros de control. En esencia, provee un método uniforme de acceso a los parámetros de banda base. El HCI está desarrollado en tres secciones, el host, la capa de transporte y el controlador del host. Cada una de las secciones tiene un rol diferente a desempeñar en este sistema.

1.5.5.1.3 L2CAP

El protocolo L2CAP (logical link control and adaptation layer protocol) está situado sobre el protocolo de banda base, estando residente en la capa de enlace de datos. Este protocolo provee servicios orientados a conexión y no orientados a conexión a los protocolos de capas superiores, incorporando capacidades de multiplexado de datos, segmentación, reunión y abstracciones de grupo. Permite a las capas superiores enviar y recibir paquetes de datos L2CAP de hasta 64KB. Esta especificación está definida únicamente para enlaces ACL, mientras que no provee soporte para enlaces SCO.

1.5.5.1.4 RFCOMM

¹⁸ <http://foro.elhacker.net/index.php/topic,83967.0.html>

El protocolo RFCOMM provee emulación de puertos serie sobre el protocolo L2CAP. Está basado en el estándar ETSI TS 07.10, del cual sólo se utilizan un subconjunto de especificaciones, además de haber adaptado una serie de las mismas.

1.5.5.1.5 SDP

El protocolo de descubrimiento de servicio (SDP) provee un conjunto de aplicaciones con el fin de descubrir qué servicios están disponibles y determinar las características de esos servicios.

En el entorno Bluetooth se necesita un protocolo específico de este tipo, ya que el repertorio de servicios disponibles cambian dinámicamente basándose en la proximidad radioeléctrica de los dispositivos en movimiento, cualitativamente diferentes a los servicios de descubrimiento de las redes convencionales. Así, el SDP definido para Bluetooth debe estar enfocado a las características únicas del entorno Bluetooth.

1.5.6 Topología de conexión de las WPAN Bluetooth.¹⁹

1.5.6.1 Picoredes WPAN Bluetooth.

Una picored es una WPAN formada por dispositivos Bluetooth que sirven como maestros en la picored y uno o más dispositivos Bluetooth que sirven de esclavos. Un canal de frequency-hopping basado en la dirección del maestro define cada picored.

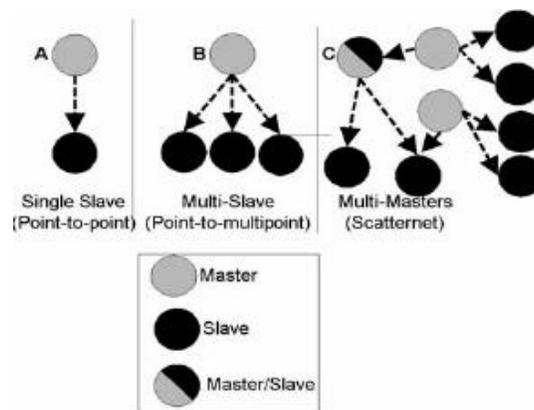
Todos los elementos participantes en una comunicación dentro de una picored dada son sincronizados al canal de frequency-hopping, utilizando el reloj del aparato maestro de la picored. Los dispositivos esclavos solo se comunican con el maestro en una estructura punto a punto bajo el control del maestro. Las transmisiones del maestro deben de ser ya sea de la forma punto a punto o bien punto a multipunto. Cada lugar determina que si algunos elementos deben ser siempre maestros o esclavos. Sin embargo, este estándar no distingue entre elementos que son permanentemente maestros o esclavos. Durante una sección de comunicación un elemento esclavo puede comportarse como maestro en un momento dado y viceversa.

¹⁹ http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/archundia_p_fm/capitulo3.pdf

1.5.6.2 Redes dispersas WPAN Bluetooth.

Una red dispersa (scatternet) es una colección de varias picoredes en operación que se traslapan en tiempo y en espacio. Un aparato Bluetooth puede participar en múltiples picoredes al mismo tiempo, por lo que existe la posibilidad de que la información fluya más allá de las fronteras de una picored. Un aparato en una scatternet puede ser esclava en varias picoredes, pero puede ser maestra en solo una de ellas. La figura 10 muestra las diferentes formas en que los dispositivos Bluetooth se pueden interconectar para formar varios sistemas de comunicación.

Figura 10. Topología de una red dispersa Bluetooth WPAN



Fuente: http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/archundia_p_fm/capitulo3.pdf

1.6 ESTRUCTURA DEL PROYECTO

1.6.1 Planteamiento

Diseño y construcción de un dispositivo que permita la supervisión de la prueba de consumo de sacarosa en ratas.

1.6.2 Objetivo General

Diseñar y construir un dispositivo que permita la supervisión de la prueba de consumo de sacarosa en ratas.

1.6.3 Objetivos Específicos

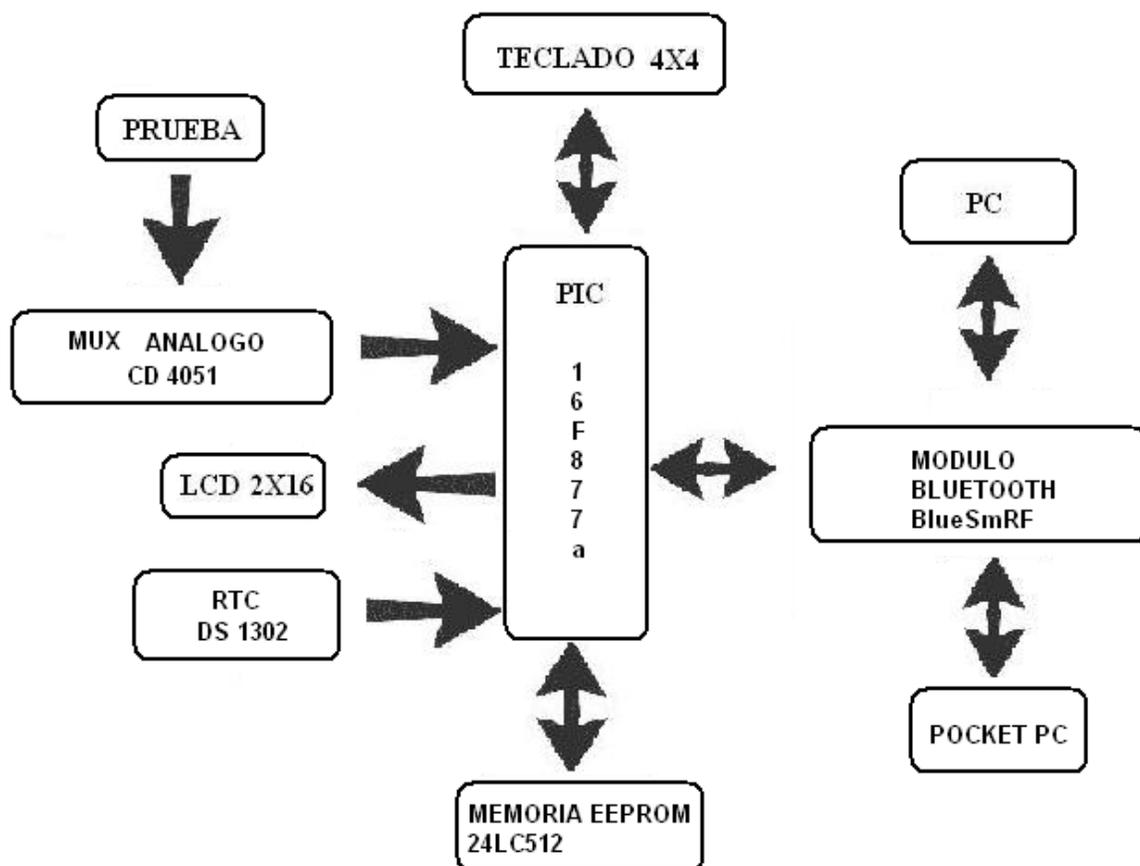
- Estudiar las características propias de la prueba de consumo de sacarosa en ratas
- Diseñar y construir la estructura física del dispositivo de supervisión para la prueba de consumo de sacarosa en ratas
- Diseñar e implementar una interfaz de usuario en la parte frontal del dispositivo en donde se pueda configurar el mismo
- Diseñar una aplicación en LabVIEW para la supervisión de la prueba de consumo de sacarosa en forma inalámbrica
- Diseñar un sistema de memoria en hardware que permita almacenar los datos adquiridos, cuando no se cuente con un PC
- Diseñar una aplicación en LabVIEW que permita la supervisión de la prueba de consumo de sacarosa en ratas desde la PocketPC

2. MÓDULOS DEL DISPOSITIVO

2.1 DIAGRAMA DE BLOQUES

A continuación se muestra en la figura 11 el diagrama de bloques correspondiente al dispositivo de supervisión de la prueba de consumo de sacarosa en ratas.

Figura 11. Diagrama de bloques del proyecto



El dispositivo de supervisión de la prueba de consumo de sacarosa posee tres modos de manejo:

- Modo directo
- Modo PC
- Modo PDA

En el modo directo el dispositivo tiene la capacidad de supervisar la prueba de consumo de sacarosa en ratas si la necesidad de utilizar un PC, esto hace que el dispositivo sea más independiente y versátil.

El modo directo cuenta con una interfaz de usuario basada en la utilización de un LCD 2X16 para la visualización y un teclado matricial 4X4 para la configuración y el ingreso de datos al dispositivo. A través del teclado matricial 4X4 se ingresan los datos de configuración o de control al dispositivo que se visualizan en el LCD 2X16, el elemento de control utilizado es un PIC 16F877a y quien es el encargado de ejecutar las acciones que configura el usuario.

El dispositivo en modo directo cuenta con un reloj de tiempo real DS1302 externo que le permite al usuario elegir tres tiempos de duración para la prueba, además de esto el dispositivo cuenta con una memoria EEPROM 24LC512 externa que permite el almacenamiento de los datos de múltiples prueba, para descargar los datos del dispositivo es necesario el uso de un PC.

Para la comunicación con la memoria EEPROM 24LC512 y el PIC 16F877a se utilizó el protocolo de comunicación síncrona I2C que es compatible para ambos dispositivos.

Tanto para el modo directo como el modo PC con el dispositivo se pueden supervisar hasta 24 jaulas cada una de ellas con dos sensores es decir se pueden adquirir un máximo de 48 señales, para la adquisición de las señales se utiliza el módulo A/D del PIC16F877a, se usan 6 canales del A/D del PIC y a su vez se utilizan multiplexores analógicos CD4051 de 8 entradas conectados a cada canal con lo que se logran las 48 señales, el control de la multiplicación de la señales las realiza el PIC.

En el modo PC el dispositivo se configura a través de una interfaz de usuario desarrollada en LabVIEW, para la comunicación con el PC el PIC se comunica mediante protocolo serial asíncrono RS232 con un Modem Bluetooth - BlueSMiRF RP-SMA que a su vez se comunica con el PC usando la especificación Bluetooth y mediante el protocolo de comunicación asíncrona RS232. En este modo el usuario tiene la posibilidad de elegir tres tiempos de duración para la prueba, además al finalizar la prueba se genera un reporte de Excel que puede ser almacenado por el usuario para su posterior análisis.

El modo PC cuenta con la opción de lectura de los datos que tenga almacenados el dispositivo en su memoria EEPROM y también se genera un reporte de Excel con los datos provenientes del dispositivo.

Otra forma de control del dispositivo es el modo PDA el cual funciona mediante el uso de una PocketPC en este modo el usuario tiene la posibilidad de supervisar 8 jaulas y las mismas opciones de configuración que el modo PC, los protocolos de

comunicación utilizados son los mismos que en el modo PC, se utiliza una interfaz de usuario especialmente diseñada en LabVIEW para la PocketPC.

El dispositivo de supervisión está compuesto por de 6 módulos de hardware:

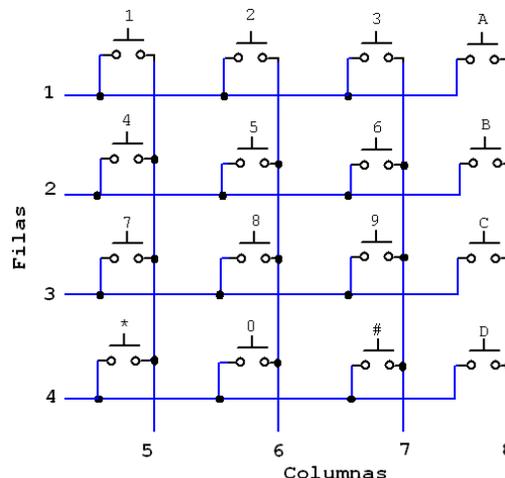
- Módulo teclado matricial 4X4.
- Módulo LCD 2X16.
- Módulo memoria EEPROM 24LC512.
- Módulo de RTC DS1302.
- Módulo comunicación Bluetooth.
- Módulo de multiplexores analógicos.
- Módulo de A/D.

La integración de los módulos de hardware mencionados anteriormente conforma el dispositivo de supervisión de la prueba de consumo de sacarosa en ratas. A continuación se detallara cada uno de estos módulos.

2.2 MÓDULO TECLADO MATRICIAL 4X4

Un teclado matricial en un simple arreglo de 16 botones conectados en filas y columnas, de modo que solo cuenta con 8 pines 4 de filas y 4 de columnas con lo cual se puede leer varios botones con el mínimo número de pines requeridos (ver figura 12).

Figura 12. Esquema circuital de un teclado matricial 4X4



2.2.1 Manejo del teclado matricial 4X4

Un teclado matricial 4x4 solamente ocupa 4 líneas de un puerto para las filas y otras 4 líneas para las columnas, de este modo se pueden leer 16 teclas utilizando solamente 8 líneas de un microcontrolador (ver figura 13) en este caso los puertos del μC PIC utilizados son los correspondientes al puerto D.

El esquema de conexiones de la figura 13 muestra la forma en la que se conectó el teclado matricial 4X4 y el microcontrolador PIC 16F877a en el dispositivo de supervisión para funcionar como interfaz de manejo los puertos RD0 a RD3 del PIC 16F877a se utilizaron para el control de las filas del teclado y los puertos RD4 a RD7 de PIC se utilizaron para el control de las columnas.

Figura 13. Diagrama circuital del teclado matricial 4X4

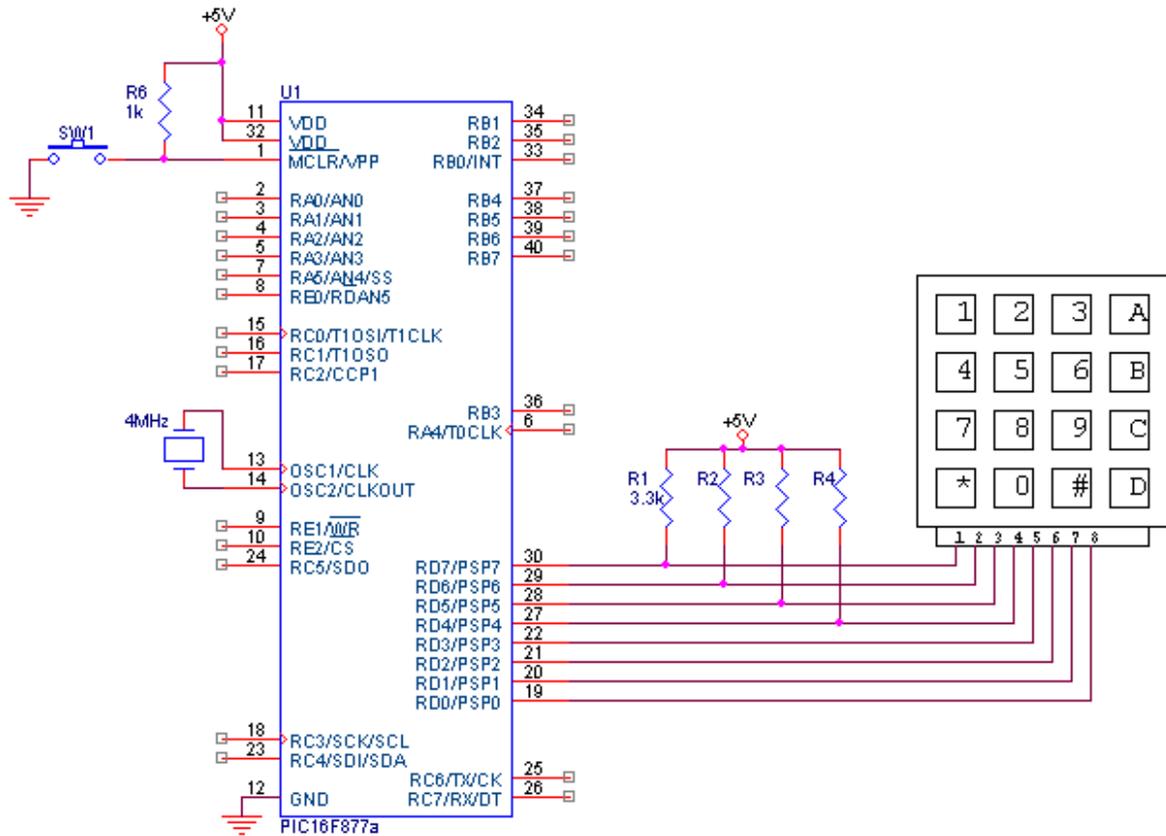
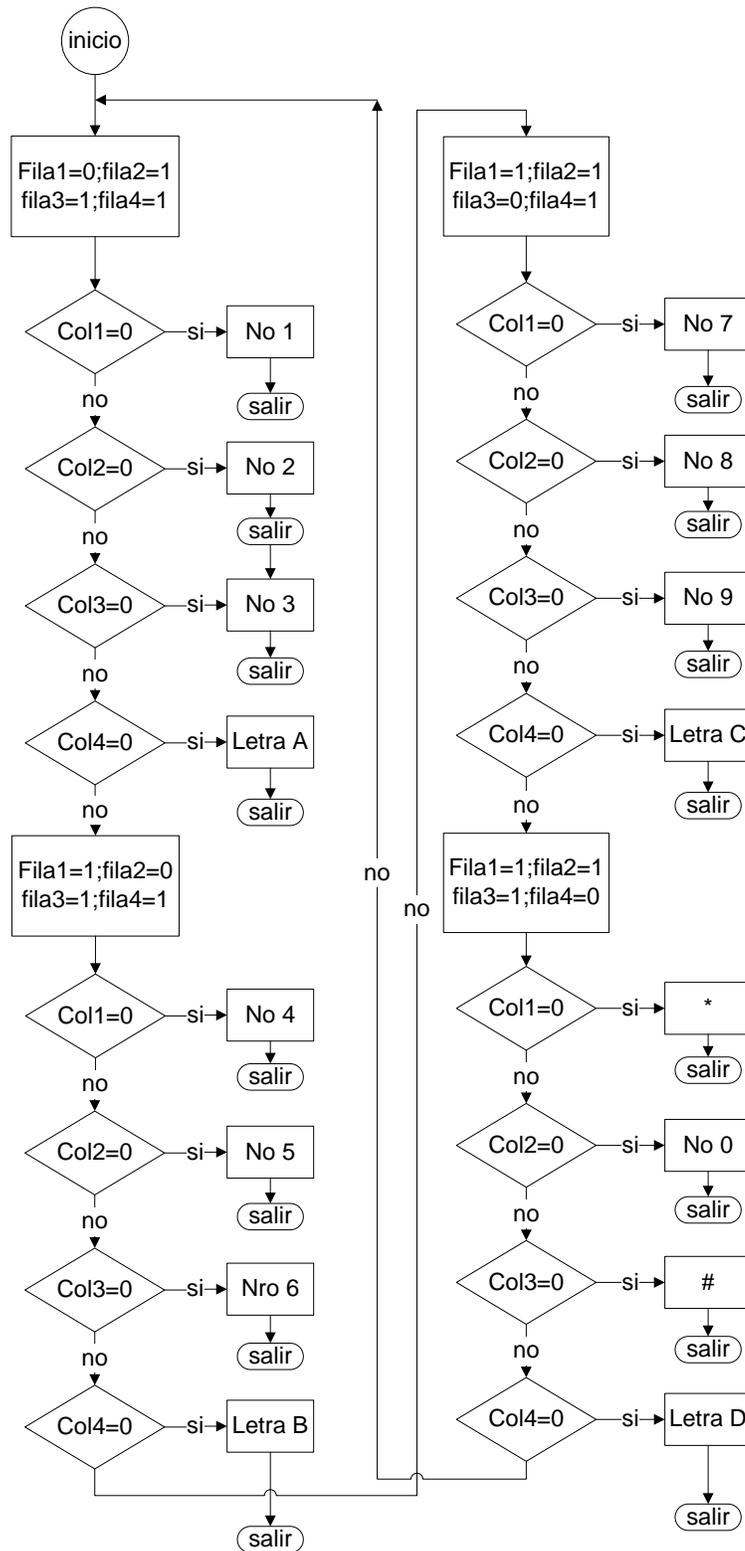


Figura 14. Diagrama de flujo de la rutina de manejo del teclado matricial 4X4



El diagrama de flujo de la figura 14 representa el algoritmo implementado en el dispositivo de supervisión para manejar un teclado matricial 4X4. Asumiendo que todas las columnas correspondientes a los puertos RD0 a RD3 del PIC y filas correspondientes a los puertos RD7 a RD4 están inicialmente en valor lógico alto (5 VDC), la pulsación de un botón se detecta de la siguiente manera: se pone en valor lógico bajo (0 VDC) el puerto RD3 correspondiente a la fila 1 y los puertos RD2 a RD0 correspondientes a las demás filas en valor lógico alto, luego se chequea cada columna (puertos RD7 a RD4) en busca de un valor lógico bajo, si alguna columna se encuentra en bajo se envía a comprobar la detección del número correspondiente pero si por el contrario no se detecta ninguna columna en bajo se vuelve a poner en uno lógico la fila 1 y a su vez se pone en bajo la fila 2 y se vuelve a chequear cada columna, si no se detecta ninguna tecla pulsada se pasa a alto la fila 2 y se coloca en bajo la fila 3 y se chequean nuevamente las columnas, si no se encuentra ninguna columna pulsada se ubica nuevamente en alto la fila 3 y la fila 4 se pasa a bajo, si no se observa alguna columna en bajo se reinicia el ciclo y se repite hasta que se pulse una tecla.

En la tabla 1 se muestra la rutina que fue programada en PIC16F877a para controlar un teclado matricial 4X4 para que este sirva como interfaz de manejo del dispositivo de supervisión.

Tabla 1. Rutina en C para el manejo del teclado matricial 4X4

```
void teclado()
{
ciclo1:
temp=0;
x=0;
col1=0;
col2=0;
col3=0;
col4=0;
temp=0;
goto teclas;

teclas:
//fila1
output_low(pin_d3);
output_high(pin_d2);
output_high(pin_d1);
output_high(pin_d0);

col1 = input(pin_d7);
col2 = input(pin_d6);
```

```

col3 = input(pin_d5);
col4 = input(pin_d4);

if(col1==0)
{
goto uno;
}
if(col2==0)
{
goto dos;
}
if(col3==0)
{
goto tres;
}
if(col4==0)
{
goto letraA;
}

//fila2
output_high(pin_d3);
output_low(pin_d2);
output_high(pin_d1);
output_high(pin_d0);

col1 = input(pin_d7);
col2 = input(pin_d6);
col3 = input(pin_d5);
col4 = input(pin_d4);

if(col1==0)
{
goto cuatro;
}
if(col2==0)
{
goto cinco;
}
if(col3==0)
{
goto seis;
}
if(col4==0)
{

```

```

        goto letraB;
    }

//fila3
output_high(pin_d3);
output_high(pin_d2);
output_low(pin_d1);
output_high(pin_d0);

col1 = input(pin_d7);
col2 = input(pin_d6);
col3 = input(pin_d5);
col4 = input(pin_d4);

if(col1==0)
{
    goto siete;
}
if(col2==0)
{
    goto ocho;
}
if(col3==0)
{
    goto nueve;
}
if(col4==0)
{
    goto letraC;
}

//fila4
output_high(pin_d3);
output_high(pin_d2);
output_high(pin_d1);
output_low(pin_d0);

col1 = input(pin_d7);
col2 = input(pin_d6);
col3 = input(pin_d5);
col4 = input(pin_d4);

if(col1==0)
{
    goto asterisco;
}

```

```
}  
if(col2==0)  
{  
    goto cero;  
}  
if(col3==0)  
{  
    goto numeral;  
}  
if(col4==0)  
{  
    goto letraD;  
}  
goto ciclo1;
```

```
uno:  
rebote();  
numero=1;  
ascii=0x31;  
goto ciclo;
```

```
dos:  
rebote1();  
numero=2;  
ascii=0x32;  
goto ciclo;
```

```
tres:  
rebote2();  
numero=3;  
ascii=0x33;  
goto ciclo;
```

```
letraA:  
rebote3();  
numero=11;  
ascii=0x41;  
goto ciclo;
```

```
cuatro:  
rebote();  
ascii=0x34;  
numero=4;  
goto ciclo;
```

```
cinco:
rebote1();
numero=5;
ascii=0x35;
goto ciclo;

seis:
rebote2();
numero=6;
ascii=0x36;
goto ciclo;

letraB:
rebote3();
numero=22;
ascii=0x42;
goto ciclo;

siete:
rebote();
numero=7;
ascii=0x37;
goto ciclo;

ocho:
rebote1();
numero=8;
ascii=0x38;
goto ciclo;

nueve:
rebote2();
numero=9;
ascii=0x39;
goto ciclo;

letraC:
rebote3();
numero=33;
ascii=0x43;
goto ciclo;

asterisco:
rebote();
numero=41;
```

```

ascii=0x2a;
goto ciclo;

cero:
rebote1();
numero=0;
ascii=0x30;
goto ciclo;

numeral:
rebote2();
numero=43;
ascii=0x23;
goto ciclo;

letraD:
rebote3();
numero=44;
ascii=0x44;
goto ciclo;

ciclo:temp=0;

}//-----fin función-----

```

2.3 MÓDULO LCD 2X16

Un LCD es un dispositivo de visualización gráfica que cuenta con una unidad μ Controlada, dispone de 2 filas de 16 caracteres cada una y cada caracter dispone de una matriz de 5x7 puntos (pixel), aunque los hay de otro número de filas y caracteres.²⁰

²⁰ <http://www.forosdeelectronica.com>

Figura 15. Fotografía de un LCD 2X16



Existen dos formas de enviar datos a un LCD, una manera es utilizando un bus de datos de 8 bits (ver figura 17) y la otra es utilizando uno de 4 bits (ver figura 18), en la conexión de 4 bits se multiplexa el bus enviando primero los 4 bits más significativos del carácter ASCII y luego los menos significativos, esto se realiza mediante programación, cabe resaltar que en la conexión circuital solo se conectan los 4 bits más significativos como se muestra en la figura 18.

Figura 16. Conexión del LCD con bus de 8 bits

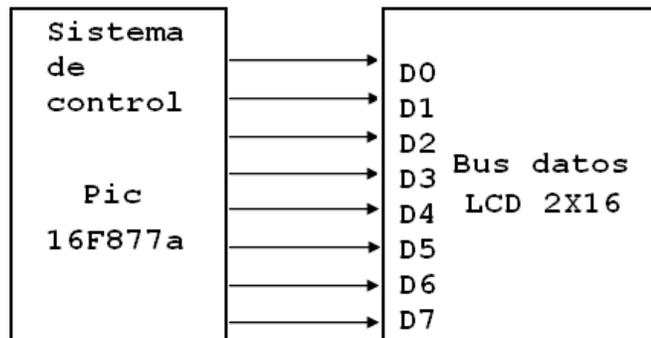
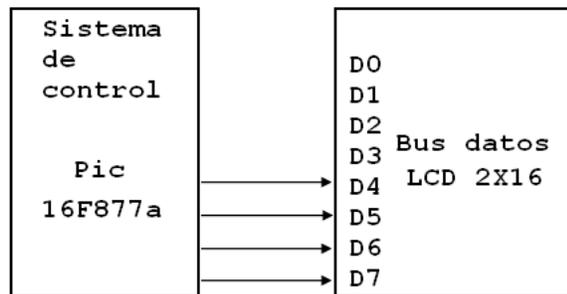
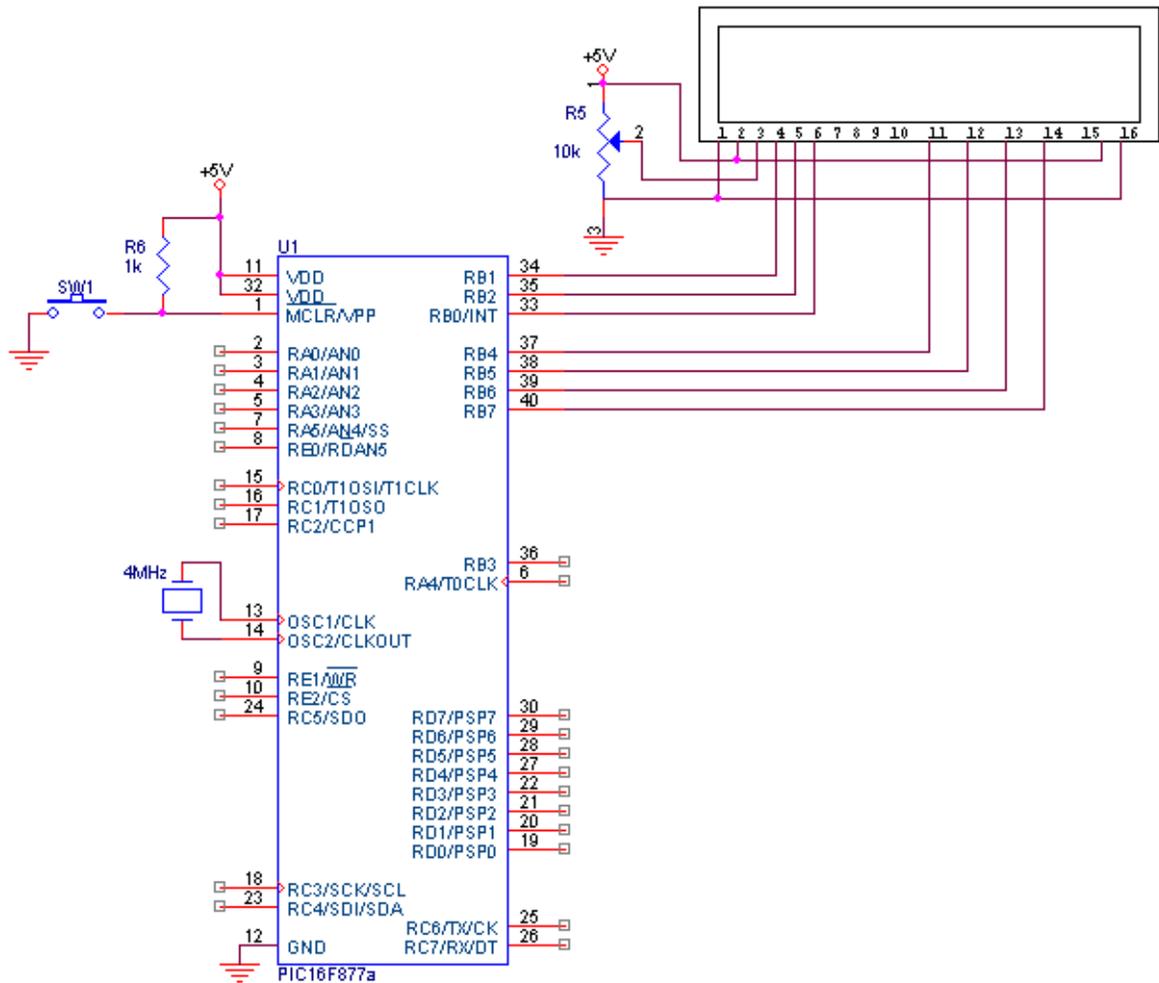


Figura 17. Conexión del LCD con bus de 4 bits



2.3.1 Manejo del LCD 2X16

Figura 18. Diagrama circuital de LCD 2X16



El dispositivo de supervisión utiliza una conexión de bus de 4 bits entre el LCD 2X16 y el PIC 16F877a como se indica en la figura 18. Para los bits de datos se utilizan el puerto RB4 a RB7, en cuanto a los bits de control se usan los puertos RB1 como RS o selección de registro de control/datos, el puerto RB2 como R/W o bit de señal de lectura/escritura y el puerto RB0 como E o bit de habilitación del módulo. Por otra parte los pin 1 del LCD 2X16 corresponde a la tierra de alimentación, el pin 2 es la alimentación positiva (+5 VDC), el pin 3 controla el contraste del LCD, y los pines 15 y 16 corresponden al control de iluminación del LCD siendo +5DCV y 0VDC respectivamente.

En el dispositivo de supervisión el LCD cumple la función de presentar a usuario los menús que le permiten acceder las funciones para las que fue hecho el

mismo, para logra la visualización de dichos menú se utilizó la librería <lcd.c> tomada del compilador *CSS C Compiler*, permite mediante comandos tipo lenguaje c de programación el control de la visualización de mensajes en el LCD. La tabla presenta la rutina de la librería <lcd.c>

Tabla 2. Rutina lcd.c, tomada de los drivers de CSS C compiler

```

////////////////////////////////////
////          LCDD.C          ////
////      Driver for common LCD modules          ////
////          ////
//// lcd_init() Must be called before any other function.          ////
////          ////
//// lcd_putc(c) Will display c on the next position of the LCD.    ////
////          The following have special meaning:          ////
////          \f Clear display          ////
////          \n Go to start of second line          ////
////          \b Move back one position          ////
////          ////
//// lcd_gotoxy(x,y) Set write position on LCD (upper left is 1,1)  ////
////          ////
//// lcd_getc(x,y) Returns character at position x,y on LCD          ////
////          ////
////////////////////////////////////
////      (C) Copyright 1996,2003 Custom Computer Services          ////
////      This source code may only be used by licensed users of the CCS C   ////
////      compiler. This source code may only be distributed to other   ////
////      licensed users of the CCS C compiler. No other use, reproduction ////
////      or distribution is permitted without written permission.      ////
////      Derivative programs created using this software in object code  ////
////      form are not restricted in any way.          ////
////////////////////////////////////

// As defined in the following structure the pin connection is as follows:
//  D0 enable
//  D1 rs
//  D2 rw
//  D4 D4
//  D5 D5
//  D6 D6
//  D7 D7
//
// LCD pins D0-D3 are not used and PIC D3 is not used.

// Un-comment the following define to use port B

```

```

#define use_portb_lcd TRUE

struct lcd_pin_map {          // This structure is overlaid
    BOOLEAN enable;          // on to an I/O port to gain
    BOOLEAN rs;              // access to the LCD pins.
    BOOLEAN rw;              // The bits are allocated from
    BOOLEAN unused;          // low order up. ENABLE will
    int data : 4;            // be pin B0.
} lcd;

#if defined(__PCH__)
#if defined use_portb_lcd
    #byte lcd = 0xF81          // This puts the entire structure
#else
    #byte lcd = 0xF83          // This puts the entire structure
#endif
#else
#if defined use_portb_lcd
    #byte lcd = 6              // on to port B (at address 6)
#else
    #byte lcd = 8              // on to port D (at address 8)
#endif
#endif

#if defined use_portb_lcd
    #define set_tris_lcd(x) set_tris_b(x)
#else
    #define set_tris_lcd(x) set_tris_d(x)
#endif

#define lcd_type 2           // 0=5x7, 1=5x10, 2=2 lines
#define lcd_line_two 0x40    // LCD RAM address for the second line

BYTE const LCD_INIT_STRING[4] = {0x20 | (lcd_type << 2), 0xc, 1, 6};
    // These bytes need to be sent to the LCD
    // to start it up.

    // The following are used for setting
    // the I/O port direction register.

```

```

struct lcd_pin_map const LCD_WRITE = {0,0,0,0,0}; // For write mode all pins are
out
struct lcd_pin_map const LCD_READ = {0,0,0,0,15}; // For read mode data pins
are in

BYTE lcd_read_byte() {
    BYTE low,high;
    set_tris_lcd(LCD_READ);
    lcd.rw = 1;
    delay_cycles(1);
    lcd.enable = 1;
    delay_cycles(1);
    high = lcd.data;
    lcd.enable = 0;
    delay_cycles(1);
    lcd.enable = 1;
    delay_us(1);
    low = lcd.data;
    lcd.enable = 0;
    set_tris_lcd(LCD_WRITE);
    return( (high<<4) | low);
}

void lcd_send_nibble( BYTE n ) {
    lcd.data = n;
    delay_cycles(1);
    lcd.enable = 1;
    delay_us(2);
    lcd.enable = 0;
}

void lcd_send_byte( BYTE address, BYTE n ) {

    lcd.rs = 0;
    while ( bit_test(lcd_read_byte(),7) );
    lcd.rs = address;
    delay_cycles(1);
    lcd.rw = 0;
    delay_cycles(1);
    lcd.enable = 0;
    lcd_send_nibble(n >> 4);
}

```

```

    lcd_send_nibble(n & 0xf);
}

void lcd_init() {
    BYTE i;
    set_tris_lcd(LCD_WRITE);
    lcd.rs = 0;
    lcd.rw = 0;
    lcd.enable = 0;
    delay_ms(15);
    for(i=1;i<=3;++i) {
        lcd_send_nibble(3);
        delay_ms(5);
    }
    lcd_send_nibble(2);
    for(i=0;i<=3;++i)
        lcd_send_byte(0,LCD_INIT_STRING[i]);
}

void lcd_gotoxy( BYTE x, BYTE y) {
    BYTE address;

    if(y!=1)
        address=lcd_line_two;
    else
        address=0;
    address+=x-1;
    lcd_send_byte(0,0x80|address);
}

void lcd_putc( char c) {
    switch (c) {
        case '\f' : lcd_send_byte(0,1);
                    delay_ms(2);
                    break;
        case '\n' : lcd_gotoxy(1,2);    break;
        case '\b' : lcd_send_byte(0,0x10); break;
        default  : lcd_send_byte(1,c);   break;
    }
}

char lcd_getc( BYTE x, BYTE y) {
    char value;

```

```
    lcd_gotoxy(x,y);  
    while ( bit_test(lcd_read_byte(),7) ); // wait until busy flag is low  
    lcd.rs=1;  
    value = lcd_read_byte();  
    lcd.rs=0;  
    return(value);  
}
```

2.4 MÓDULO MEMORIA EEPROM 24LC512

El dispositivo de supervisión de la prueba de consumo de sacarosa en ratas utiliza una memoria externa EEPROM 24LC512 para el almacenamiento de los datos que arroja la prueba cuando se trabaja con el dispositivo en modo directo, la memoria es del tipo serie 64K x 8 es decir que pueden escribir o borrar eléctricamente 512Kbits. Esta memoria cuenta con una interfaz de comunicación seria I2C, dicha interfaz se utilizo para establecer comunicación entre el PIC 16F877a que funciona como el dispositivo maestro y la memoria 24LC512 que trabaja como dispositivo esclavo.

Cabe resaltar que a pesar de que el PIC 16F877a posee una memoria EEPROM interna esta no se utilizó ya que el número de datos que se requerían almacenar eran superaban ampliamente la capacidad de almacenamiento de la memoria del PIC.

- **SDA:** es un pin de entrada y salida de datos hacia y desde el dispositivo. Es un pin diseñado en colector abierto por lo cual se requiere el uso de una resistencia de pull up conectada a la alimentación positiva, el valor de la resistencia es de 10KΩ para frecuencias de 100KHz y de 2KΩ para frecuencias de 400KHz y 1Mhz. En cuanto al dispositivo de supervisión se utilizaron resistencias de pull up de 5KΩ. En el dispositivo de supervisión el pin correspondiente al RC4 del PIC 16F877a es el pin SDA que se conecta con su correspondiente en la memoria 24LC512.
- **SLC:** es un pin que se encarga de manejar el sincronismo de la transferencia de datos del dispositivo. En el dispositivo de supervisión el pin del puerto RC3 corresponde al SLC del PIC 16F877a y esta cableado con el pin SLC correspondiente de la memoria 24LC512.
- **WP:** este pin controla la función de lectura y escritura de la memoria por lo cual se requiere que esté conectado siempre en VCC o VSS es decir que no debe permanecer flotante. Si el pin WP se encuentra en un valor lógico bajo la memoria puede leer y escribir datos normalmente, pero si WP está en un valor lógico alto la función de escritura de la memoria se inhibe mientras que la función de lectura no se afecta.

2.4.2 Datos a almacenar en memoria

El dispositivo de supervisión cuando se trabaja en modo directo es controlado mediante una interfaz formada por un teclado matricial 4X4 y un LCD 2X16, en este modo almacenan los Bytes que se muestran en la tabla por cada rata.

Tabla 3. Datos almacenados por cada rata

	Numero de bytes
ID de la rata	4
% sacarosa	2
Fecha	6
Hora	4
Volumen inicial sacarosa	3
Volumen inicial agua	3
Volumen final sacarosa	3
Volumen final agua	3
Consumo de sacarosa	3
Consumo de agua	3
TOTAL	34

ID de la rata, se utiliza para identificar la rata a la que se le está aplicando la prueba.

La fecha y la hora, sirven para diferenciar los datos almacenados en el momento de descargarlos al PC.

Los bytes de volumen inicial y final de sacarosa y de agua se utilizan para calcular el consumo de agua y sacarosa de la rata.

Los bytes de consumo de agua y sacarosa son los que relaciona el comportamiento de la rata en la prueba.

Además se utilizan 3 bytes para llevar el conteo de las ratas que se han almacenado.

El número máximo de ratas que se pueden almacenar en la memoria es de 984 ratas lo cual equivale a realizar 41 pruebas de 24 ratas cada una. Es decir que el número máximo de bytes que se puede almacenar en la memoria EEPROM del dispositivo es de 33459 Bytes o 267672 bits.

2.4.3 Manejo de la memoria 24LC512

En la implementación del dispositivo de supervisión de la prueba de consumo de sacarosa en ratas, se utilizan la operación de escritura y lectura de la memoria EEPROM 24LC512, la memoria es controlada mediante la interfaz de comunicación I2C por el PIC 16F877a. A continuación se explica la forma de escribir la memoria.

2.4.3.1 Escritura de la memoria 24LC512

La escritura de la memoria 24LC512 en el dispositivo de supervisión se realiza mediante el uso de la librería <24512.c> (ver tabla 4) del compilador *CSS C Compiler* la cual se carga en el programa que está almacenado en el PIC 16F877a. Con esta librería se pueden escribir bytes mediante el uso de una función tipo lenguaje C que se explica a continuación:

write_ext_eeprom(A, B); donde A es el byte que se desea almacenar en la memoria y B es la dirección en donde se almacenará el dato A dentro de la memoria 24LC512.

En la tabla 4 se muestra la librería <24512.c>, esta librería es también usada para la lectura de la memoria, cabe resaltar que antes de utilizar la función *write_ext_eeprom(A, B)*; se debe habilitar la función mediante el comando *init_ext_eeprom()*; al inicio del programa principal que contiene el PIC 16F877a.

Tabla 4. Rutina 24512.c, tomada de los drivers de CSS C Compiler

```

////////////////////////////////////
/// Library for a 24LC512 serial EEPROM          ///
///                                           ///
/// init_ext_eeprom(); Call before the other functions are used ///
///                                           ///
/// write_ext_eeprom(a, d); Write the byte d to the address a    ///
///                                           ///
/// d = read_ext_eeprom(a); Read the byte d from the address a  ///
///                                           ///
/// The main program may define eeprom_sda          ///
/// and eeprom_scl to override the defaults below.    ///
///                                           ///
////////////////////////////////////
/// (C) Copyright 1996,2003 Custom Computer Services    ///
/// This source code may only be used by licensed users of the CCS C ///
/// compiler. This source code may only be distributed to other    ///
/// licensed users of the CCS C compiler. No other use, reproduction ///
/// or distribution is permitted without written permission.    ///
/// Derivative programs created using this software in object code ///
/// form are not restricted in any way.                ///
////////////////////////////////////

#ifndef EEPROM_SDA

#define EEPROM_SDA PIN_C4
#define EEPROM_SCL PIN_C3

#endif

#include i2c(master, sda=EEPROM_SDA, scl=EEPROM_SCL)

#define EEPROM_ADDRESS long int
#define EEPROM_SIZE 65535

void init_ext_eeprom()
{
    output_float(EEPROM_SCL);
    output_float(EEPROM_SDA);
}

void write_ext_eeprom(long int address, BYTE data)
{

```

```

short int status;
i2c_start();
i2c_write(0xa0);
i2c_write(address>>8);
i2c_write(address);
i2c_write(data);
i2c_stop();
i2c_start();
status=i2c_write(0xa0);
while(status==1)
{
    i2c_start();
    status=i2c_write(0xa0);
}
}

BYTE read_ext_eeprom(long int address) {
    BYTE data;
    i2c_start();
    i2c_write(0xa0);
    i2c_write(address>>8);
    i2c_write(address);
    i2c_start();
    i2c_write(0xa1);
    data=i2c_read(0);
    i2c_stop();
    return(data);
}

```

2.4.3.2 Lectura de la memoria 24LC512

En el modo de manejo PC del dispositivo de supervisión de la prueba de consumo de sacarosa en ratas, el usuario tiene la posibilidad de efectuar la lectura de los datos que se encuentran almacenados en la memoria EEPROM 24LC512.

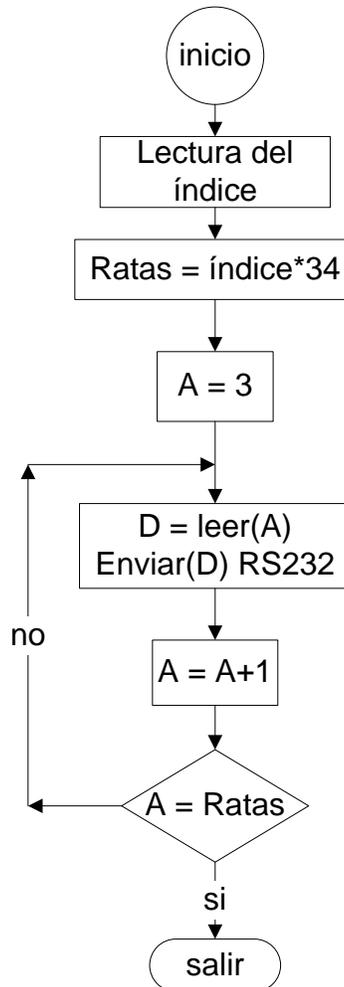
Para efectuar la operación de lectura de los datos que se encuentran almacenados en memoria 24LC512, se utiliza un PIC 16F877a como dispositivo maestro y la memoria como esclavo dentro de la interfaz de comunicación serial I2C que se utiliza para hacer la comunicación entre los dispositivos.

En el algoritmo programado en el PIC 16F877a se utiliza la función $D = read_ext_eeprom(A)$; a la cual se encuentra programada en la librería <24512.c> (ver tabla 4) de CSS C Compiler, la cual debe ser cargada en el programa del PIC.

Cabe resaltar que la función de escritura de la memoria 24LC512 también se encuentra en la librería <24512.c>

En la función $D = read_ext_eeprom(A)$; el valor de A corresponde a la dirección de la memoria que se desea leer y D es la variable en donde se almacena el valor leído de la memoria en la dirección A.

Figura 20. Diagrama de flujo para la lectura de los datos de la memoria 24LC512



En el diagrama de bloques de la figura 20, inicialmente se lee el valor del índice es decir el número de ratas que se han almacenado en la memoria y que se van a leer, después se determina el número de bytes que se van a leer teniendo en cuenta el número de bytes y que por cada rata se almacena es 34 bytes (ver tabla 3), luego se procede a leer los datos desde la dirección 3 de la memoria y además se envía el dato leído al modem de comunicación Bluetooth BlueSMiRF RP-SMA mediante el protocolo RS232.

La tabla 5 presenta el algoritmo implementado en el lenguaje C para el PIC 16F877a para realizar la función de lectura de los datos almacenados en la memoria 24LC512

Tabla 5. Rutina en C para la lectura de la memoria 24LC512

```
void leer1(){
lcd_putc("\fEnviando datos...\n");
lcd_putc("\n");
delay_ms(2000);
contador=0;
temp1=0;
r=0;
num1=0;

r1 = read_ext_eeprom(contador); // lectura del índice
r1=r1-0x30;
r1=r1*0x64;
contador++;

h1 = read_ext_eeprom(contador); //lectura del índice
h1=h1-0x30;
h1=h1*0x0A;
contador++;

h = read_ext_eeprom(contador); //lectura del índice
h=h-0x30;
h=h+h1+r1;

delay_ms(100);
num1=h;
contador=0;

entrar1:
putc(0x7A);
entrada=getch();
if(entrada==0x7A)
    {
        goto segir1;
    }
goto entrar1;

segir1:
```

```

do
    {
        r = read_ext_eeprom(contador);    //#ratas
        putc(r);
        delay_ms(5);
        contador++;
    }
while(contador<3);

delay_ms(5);
contador=0;
num1=num1*34+2;
temp1=2;//12

lectura1:                                //lectura de consumo
lcd_putc("\n");
do
    {
        temp1++;
        r = read_ext_eeprom(temp1);    //lectura ID rata
        putc(r);
        delay_ms(5);
        contador++;
    }
while(contador<4);

delay_ms(5);

do
    {
        temp1++;
        r = read_ext_eeprom(temp1);    //sacarosa
        putc(r);
        delay_ms(5);
        contador++;
    }
while(contador<6);

delay_ms(5);

do
    {
        temp1++;
        r = read_ext_eeprom(temp1);    //fecha de inicio1
        putc(r);

```

```

        lcd_putc(r);
        delay_ms(5);
        contador++;
    }
while(contador<8);

delay_ms(5);

do
    {
        temp1++;
        r = read_ext_eeprom(temp1);    //fecha de inicio2
        putc(r);
        lcd_putc(r);
        delay_ms(5);
        contador++;
    }
while(contador<10);
delay_ms(5);

do
    {
        temp1++;
        r = read_ext_eeprom(temp1);    //fecha de inicio3
        putc(r);
        lcd_putc(r);
        delay_ms(5);
        contador++;
    }
while(contador<12);

delay_ms(5);

do
    {
        temp1++;
        r = read_ext_eeprom(temp1);    //hora1
        putc(r);
        delay_ms(5);
        contador++;
    }
while(contador<14);
delay_ms(5);

do

```

```

    {
    temp1++;
    r = read_ext_eeprom(temp1);    //hora2
    putc(r);
    delay_ms(5);
    contador++;
    }
while(contador<16);

delay_ms(5);
temp1=temp1+12;

do
    {
    temp1++;
    r = read_ext_eeprom(temp1);    //lectura sacarosa
    putc(r);
    delay_ms(5);
    contador++;
    }
while(contador<19);

    delay_ms(5);
do
    {
    temp1++;
    r = read_ext_eeprom(temp1);    //lectura agua
    putc(r);
    delay_ms(5);
    contador++;
    }
while(contador<22);

delay_ms(5);

if(temp1==num1)
    {
    goto salida_le;
    }
else
    {
    contador=0;
    goto lectura1;
    }

```

```
salida_le:
temp=temp;

}//-----fin funcion-----
```

2.5 MÓDULO RTC DS1302

El dispositivo de supervisión utiliza un contador de tiempo real con el fin que la duración de la prueba pueda ser configurada por el usuario cuando el dispositivo se trabaja en el modo directo, para este efecto se uso un contador de tiempo real (RTC) DS1302 conectado con el PIC 16F877a.

El DS1302 es un RTC/CALENDARIO, este dispositivo se comunica con el PIC 16F877a a través de una interfaz de comunicación serial síncrona simple, el DS1302 ofrece la fecha expresada en horas, minutos, segundos y la fecha expresada en día, mes, año. El reloj se auto configura al terminar cada mes y cada año. Cabe resaltar que el RTC es un dispositivo diseñado especialmente para funcionar bajo condiciones de bajo consumo de potencia y que requiere del uso de una batería que lo mantenga alimentado continuamente.

2.5.1 Descripción de los pines²²

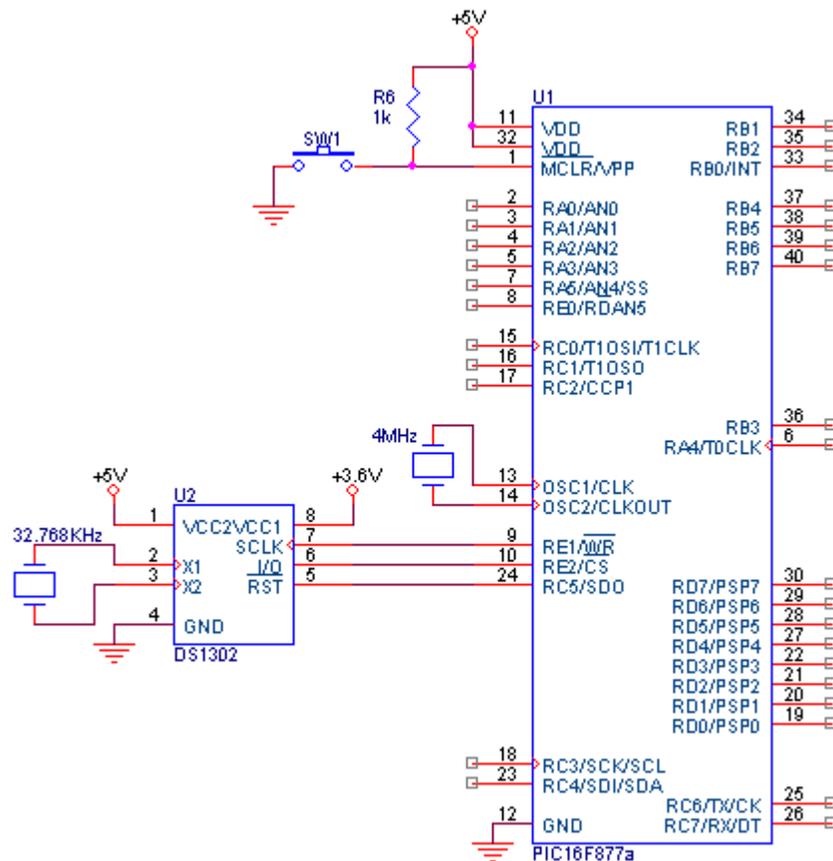
El DS1302 cuenta con 8 pines de conexión cuyas funciones dentro del circuito del dispositivo de supervisión se describen a continuación teniendo en cuenta la figura22:

- **VCC1:** es un pin de alimentación de baja potencia al que se conecta una batería de respaldo que se encarga de alimentar al DS1302 mientras la alimentación principal no está presente.
- **VCC2:** es el pin de alimentación principal de RTC que está configurado en doble oferta de alimentación, es decir si falla VCC2 entra VCC1 a trabajar VCC1 y de esta forma garantizar la fecha y la hora.
- **SCLK:** es el pin de la entrada del reloj serial, su función es sincronizar el envío de datos a través de la interfaz de comunicación serial. En el dispositivo de supervisión el pin SCLK del DS1302 está conectado con el pin RE1 del PIC 16F877a.

²² ²² <http://datasheets.maxim-ic.com/en/ds/DS1302.pdf>

- **I/O:** es el pin de entrada y salida de datos del DS1302, para el caso particular del dispositivo de supervisión este pin está conectado el pin RE2 del PIC 16F877a.
- **X1, X2:** son los pines utilizados para la conexión del oscilador de cristal de cuarzo de 32.768 KHz indispensable para correcto funcionamiento del DS1302.
- **RTS:** es el pin de Reset o de restablecimiento de la señal, este pin debe permanecer en alto mientras se realiza una operación de lectura o de escritura en el DS1302. En el dispositivo de supervisión este pin esta conectado al puerto RC5 del PIC 16F877a.

Figura 21. Diagrama circuital del RTC DS1302



2.5.2 Manejo del RTC DS1302

Dado que el dispositivo de supervisión debe garantizar tres diferentes tiempos de para la duración de la supervisión de la prueba de consumo de sacarosa en ratas y además se necesita diferenciar las pruebas a medida que se guardan en memoria y una forma para hacer esto es a través de la fecha y la hora en que se realizan es por esto que se hizo necesario el uso del RTC DS 1302.

El DS1302 es controlado mediante una interfaz de comunicación serial simple por un PIC16F877a, dicha interfaz utiliza tres líneas de conexión (pines I/O, SCLK y RST) para trabajar.

Inicialmente se configuró la fecha y la hora del DS1302 a través del PIC 16F877a mediante un el uso de función *rtc_set_datetime(day,mth,year,dow,hour,min)*, esta función se encuentra dentro de la librería <DS1302.c> del compilador CSS C Compiler (ver tabla 6).

La librería <DS1302.c > contiene las funciones que permiten la configuración y lectura de la fecha y la hora del DS1302 mediante un PIC 16F877a que posee el algoritmo que permite realizar estas operaciones. Para el caso del dispositivo de supervisión el PIC lee la hora y la fecha del DS1302 al iniciarse la prueba y las almacena en memoria y además toma la hora como referencia para controlar la duración de la supervisión de la prueba de acuerdo con lo que el usuario haya elegido.

Tabla 6. Rutina DS1302.c, tomada de los drivers de CSS C Compiler

```
//////////////////////////////////////////////////////////////////
///          DS1302.C          ///
///      Driver for Real Time Clock      ///
///          ///
/// rtc_init()          Call after power up///
///          ///
/// rtc_set_datetime(day,mth,year,dow,hour,min) Set the date/time ///
///          ///
/// rtc_get_date(day,mth,year,dow)          Get the date    ///
///          ///
/// rtc_get_time(hr,min,sec)          Get the time    ///
///          ///
/// rtc_write_nvr(address,data)          Write to NVR    ///
///          ///
/// data = rtc_read_nvr(address)          Read from NVR    ///
///          ///
/// get_bcd(data)          Convert data to BCD    ///
```

```

////                                     ////
//// rm_bcd(data)                       Convert data to int ////
////                                     ////
////////////////////////////////////
//// (C) Copyright 1996,2003 Custom Computer Services      ////
//// This source code may only be used by licensed users of the CCS C ////
//// compiler. This source code may only be distributed to other ////
//// licensed users of the CCS C compiler. No other use, reproduction////
//// or distribution is permitted without written permission.  ////
//// Derivative programs created using this software in object code ////
//// form are not restricted in any way.                       ////
////////////////////////////////////

#ifndef RTC_SCLK

#define RTC_SCLK PIN_E1//PIN_B1
#define RTC_IO  PIN_E2//PIN_B3
#define RTC_RST PIN_C5//PIN_B2

#endif

void write_ds1302_byte(BYTE cmd) {
    BYTE i;

    for(i=0;i<=7;++i) {
        output_bit(RTC_IO, shift_right(&cmd,1,0) );
        output_high(RTC_SCLK);
        output_low(RTC_SCLK);
    }
}

void write_ds1302(BYTE cmd, BYTE data) {

    output_high(RTC_RST);
    write_ds1302_byte(cmd);
    write_ds1302_byte(data);
    output_low(RTC_RST);
}

BYTE read_ds1302(BYTE cmd) {
    BYTE i,data;

    output_high(RTC_RST);
    write_ds1302_byte(cmd);

```

```

for(i=0;i<=7;++i) {
    shift_right(&data,1,input(RTC_IO));
    output_high(RTC_SCLK);
    delay_us(2);
    output_low(RTC_SCLK);
    delay_us(2);
}
output_low(RTC_RST);

return(data);
}

void rtc_init() {
    BYTE x;
    output_low(RTC_RST);
    delay_us(2);
    output_low(RTC_SCLK);
    write_ds1302(0x8e,0);
    write_ds1302(0x90,0xa6);
    x=read_ds1302(0x81);
    if((x & 0x80)!=0)
        write_ds1302(0x80,0);
}

int get_bcd(BYTE data)
{
    int nibh;
    int nibl;

    nibh=data/10;
    nibl=data-(nibh*10);

    return((nibh<<4)|nibl);
}

int rm_bcd(BYTE data)
{
    int i;

    i=data;
    data=(i>>4)*10;
    data=data+(i<<4)>>4);

    return data;
}

```

```

void rtc_set_datetime(BYTE day, BYTE mth, BYTE year, BYTE dow, BYTE hr,
BYTE min) {

    write_ds1302(0x86,get_bcd(day));
    write_ds1302(0x88,get_bcd(mth));
    write_ds1302(0x8c,get_bcd(year));
    write_ds1302(0x8a,get_bcd(dow));
    write_ds1302(0x84,get_bcd(hr));
    write_ds1302(0x82,get_bcd(min));
    write_ds1302(0x80,get_bcd(0));
}

void rtc_get_date(BYTE& day, BYTE& mth, BYTE& year, BYTE& dow) {
    day = rm_bcd(read_ds1302(0x87));
    mth = rm_bcd(read_ds1302(0x89));
    year = rm_bcd(read_ds1302(0x8d));
    dow = rm_bcd(read_ds1302(0x8b));
}

void rtc_get_time(BYTE& hr, BYTE& min, BYTE& sec) {
    hr = rm_bcd(read_ds1302(0x85));
    min = rm_bcd(read_ds1302(0x83));
    sec = rm_bcd(read_ds1302(0x81));
}

void rtc_write_nvr(BYTE address, BYTE data) {
    write_ds1302(address|0xc0,data);
}

BYTE rtc_read_nvr(BYTE address) {
    return(read_ds1302(address|0xc1));
}

```

2.5.3 Lectura de la fecha y la hora

Pariendo del diagrama de flujo de la figura 23 el cual representa el algoritmo implementado en el PIC 16F877a para realizar la lectura de la fecha y la hora que entrega el DS1302, en el se muestra como se captura la fecha y la hora y luego se convierten los datos que el DS1302 entrega en hexadecimal a ASCII.

En la tabla 7 se muestra el algoritmo que tiene el PIC 16F877a programado en lenguaje C cuya función es la de capturar la fecha y la hora y hacer la conversión de estos datos a ASCII mediante una función que se muestra en la tabla 8.

Figura 22. Diagrama de flujo de la lectura de la fecha y la hora

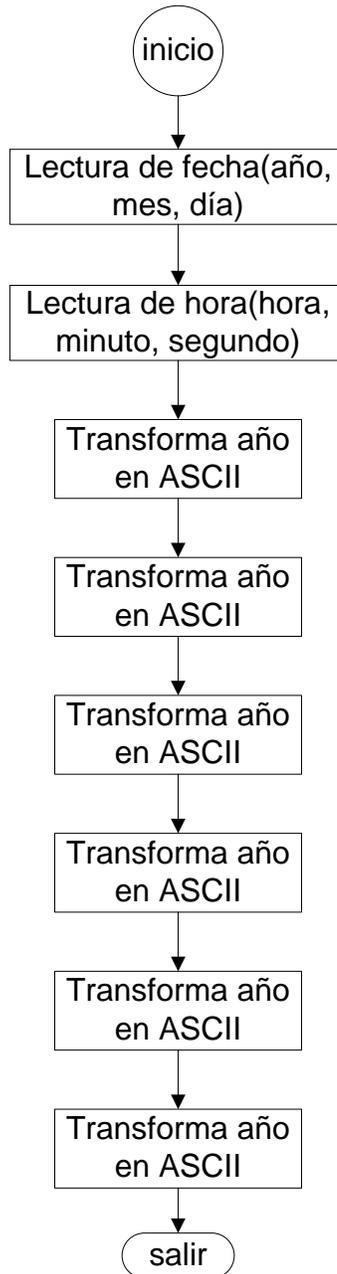


Tabla 7. Rutina en C para la lectura de la fecha y hora del DS1302

```

void calendario(){

rtc_get_date(dia,mes,anio,dow); //coge dia,mes,año
rtc_get_time(hor,min,sec ); //coge hora,minuto,segundo

hex_dec(dia);
dia=hora2;

hex_dec(mes);
mes=hora2;

hex_dec(anio);
anio=hora2;
hex_dec(hor);
hor=hora2;
hex_dec(min);
min=hora2;

}//-----fin funcion-----

```

Tabla 8. Rutina en c para conversión de los datos HEX a ASCII o decimal

```

void hex_dec(byte dato){
switch(dato){ //limites...
case 0x00: hora=0x00;hora2=0;break;
case 0x01: hora=0x01;hora2=1;break;
case 0x02: hora=0x02;hora2=2;break;
case 0x03: hora=0x03;hora2=3;break;
case 0x04: hora=0x04;hora2=4;break;
case 0x05: hora=0x05;hora2=5;break;
case 0x06: hora=0x06;hora2=6;break;
case 0x07: hora=0x07;hora2=7;break;
case 0x08: hora=0x08;hora2=8;break;
case 0x09: hora=0x09;hora2=9;break;
case 0x0A: hora=0x10;hora2=10;break;
case 0x0B: hora=0x11;hora2=11;break;
case 0x0C: hora=0x12;hora2=12;break;
case 0x0D: hora=0x13;hora2=13;break;
case 0x0E: hora=0x14;hora2=14;break;
case 0x0F: hora=0x15;hora2=15;break;
case 0x10: hora=0x16;hora2=16;break;
case 0x11: hora=0x17;hora2=17;break;

```

```

case 0x12: hora=0x18;hora2=18;break;
case 0x13: hora=0x19;hora2=19;break;
case 0x14: hora=0x20;hora2=20;break;
case 0x15: hora=0x21;hora2=21;break;
case 0x16: hora=0x22;hora2=22;break;
case 0x17: hora=0x23;hora2=23;break;
case 0x18: hora=0x24;hora2=24;break;
case 0x19: hora=0x25;hora2=25;break;
case 0x1A: hora=0x26;hora2=26;break;
case 0x1B: hora=0x27;hora2=27;break;
case 0x1C: hora=0x28;hora2=28;break;
case 0x1D: hora=0x29;hora2=29;break;
case 0x1E: hora=0x30;hora2=30;break;
case 0x1F: hora=0x31;hora2=31;break;
case 0x20: hora=0x32;hora2=32;break;
case 0x21: hora=0x33;hora2=33;break;
case 0x22: hora=0x34;hora2=34;break;
case 0x23: hora=0x35;hora2=35;break;
case 0x24: hora=0x36;hora2=36;break;
case 0x25: hora=0x37;hora2=37;break;
case 0x26: hora=0x38;hora2=38;break;
case 0x27: hora=0x39;hora2=39;break;
case 0x28: hora=0x40;hora2=40;break;
case 0x29: hora=0x41;hora2=41;break;
case 0x2A: hora=0x42;hora2=42;break;
case 0x2B: hora=0x43;hora2=43;break;
case 0x2C: hora=0x44;hora2=44;break;
case 0x2D: hora=0x45;hora2=45;break;
case 0x2E: hora=0x46;hora2=46;break;
case 0x2F: hora=0x47;hora2=47;break;
case 0x30: hora=0x48;hora2=48;break;
case 0x31: hora=0x49;hora2=49;break;
case 0x32: hora=0x50;hora2=50;break;
case 0x33: hora=0x51;hora2=51;break;
case 0x34: hora=0x52;hora2=52;break;
case 0x35: hora=0x53;hora2=53;break;
case 0x36: hora=0x54;hora2=54;break;
case 0x37: hora=0x55;hora2=55;break;
case 0x38: hora=0x56;hora2=56;break;
case 0x39: hora=0x57;hora2=57;break;
case 0x3A: hora=0x58;hora2=58;break;
case 0x3B: hora=0x59;hora2=59;
}          //...min

}//-----fin funcion-----

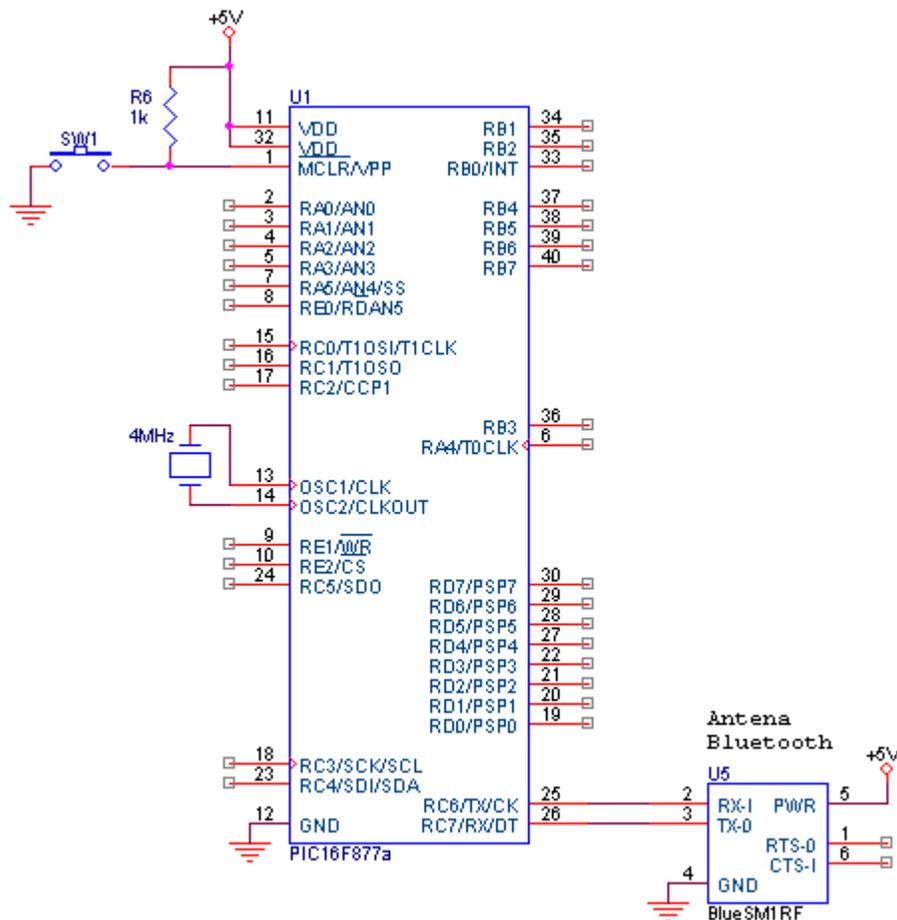
```

2.6 MÓDULO DE COMUNICACIÓN BLUETOOTH

El dispositivo de supervisión cuenta con un modo de manejo utilizando un PC y otro utilizando una PocketPC, en dichas formas de manejo la comunicación entre el dispositivo de supervisión y el PC p la PocketPC según sea el caso se realiza mediante el uso de un modem Bluetooth - BlueSMiRF RP- SMA, el cual se comunica a su vez con el PIC 16F877a a través de la interfaz de comunicación serial asíncrona RS232 que maneja el modulo USART del PIC.

2.6.1 Descripción de los pines de módulo USART y del modem Bluetooth - BlueSMiRF RP- SMA

Figura 23. Diagrama circuital del modulo Bluetooth



Los pines de PIC 16F877a usados para establecer la comunicación usando el módulo USART así como los pines del modem Bluetooth se muestran el diagrama circuital de la figura 24 y se describen a continuación:

Descripción de los pines de módulo USART:

- **RC6/TX/CK:** es el pin encargado de la transmisión de los datos y corresponde al puerto C del PIC 16F877a, en el dispositivo de supervisión este pin está conectado con el pin RX-I del modem Bluetooth - BlueSMiRF RP- SMA.
- **RC7/RX/DT:** este pin corresponde al puerto C del PIC 16F877a y su funciones la recibir los datos, en el dispositivo de supervisión este pin está conectado con el pin TX-0 del modem Bluetooth - BlueSMiRF RP- SMA.

Descripción de los pines del modem Bluetooth - BlueSMiRF RP- SMA:

- **PWR:** es el pin para la alimentación el rango de voltaje va de 3 a 10 VDC.
- **GND:** es el pin de la tierra de alimentación del circuito.
- **RX-I:** es el pin del modem Bluetooth - BlueSMiRF RP- SMA encargado de la recepción.
- **TX-0:** es el pin del modem Bluetooth - BlueSMiRF RP- SMA encargado de la transmisión de datos.
- **CTS-I:** es el pin usado limpiar el canal de envío de modem, en dispositivo no se está conectado.
- **RTS-0:** es el pin encargado de preparar el canal de envío del modem, en el dispositivo no se encuentra conectado.

2.6.2 Módulo USART del PIC 16F877a

El módulo de transmisión y recepción universal serial asíncrona (USART - Universal Synchronous Asynchronous Receiver Transmitter), es una interfaz de comunicación serial que tiene el PIC 16F877a. En el dispositivo de supervisión el USART se utiliza comunicar un modem Bluetooth – BlueSMiRF RP- SMA con el PIC 16F877a, cabe resaltar que el modem Bluetooth es quien establece la comunicación.

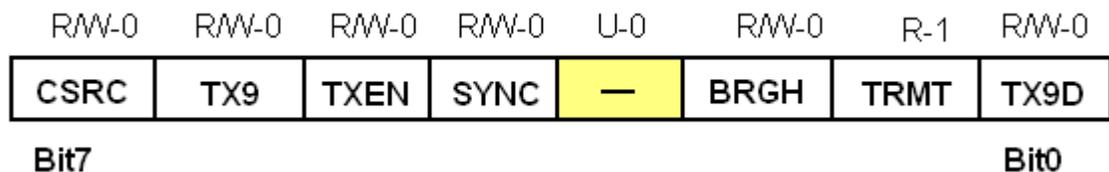
2.6.2.1 Registros del módulo USART de PIC 16F877a²³

El módulo USART posee dos registros distribuidos así:

- **TXSTA:** es el registro que controla en estado de la transmisión, su dirección es 98h.
- **RCSTA:** es el registro encargado del control del estado de la recepción, su dirección es 18h.

A continuación se describen los bits de la figura 25 los cuales que conforman la estructura de registro TXSTA.

Figura 24. Estructura del registro TXSTA



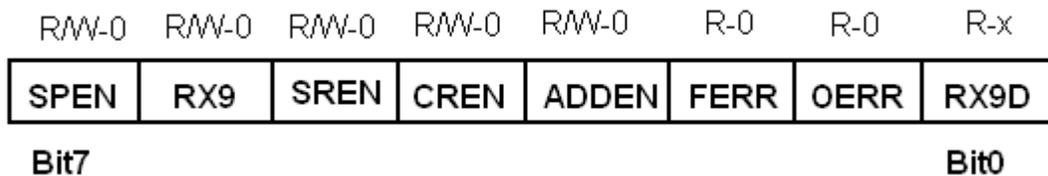
- **TXSTA:** registro de control de estado de la transmisión, es necesario tener en cuenta que si un bit tiene el encabezado R/W esto indica que puede ser leído y escrito, si tiene el encabezado U indica que no se usa y el encabezado R implica que solamente puede ser leído.
 1. **CSRC:** es el bit que selecciona la fuente de reloj, en modo asíncrono no se tiene en cuenta, por el contrario en el modo síncrono funciona así:
 - 0: modo esclavo.
 - 1: modo maestro.
 2. **TX9:** bit de activación de la transmisión de 8 o 9 bits.
 - 0: 8 bits de transmisión.
 - 1: 9 bits de transmisión.
 3. **TXEN:** bit de activación de la transmisión.
 - 0: transmisión desactivada.
 - 1: transmisión activada.

²³ <http://ww1.microchip.com/downloads/en/devicedoc/21754E.pdf>

4. **SYNC:** bit que selecciona el modo USART.
 0: modo asíncrono.
 1: modo síncrono.
5. El bit 3 no aplica y se coloca en 0.
6. **BRGH:** bit de selección de la velocidad en baudios, en modo asíncrono debe seleccionarse velocidad baja
 0: velocidad baja.
 1 velocidad alta.
7. **TRMT:** bit de estado del registro de desplazamiento de transmisión.
 0: TSR completo.
 1: TSR vacío.
8. **T9DX:** bit 9 de la transmisión de datos, se puede usar como bit de paridad.

Tomado como referencia la figura 26 se describen a continuación los bits del registro RCSTA.

Figura 25. Estructura del registro RCSTA



- **RCSTA:** es el registro que controla el estado de la recepción. Los bits con el encabezado R/W indican que son bits que pueden ser leídos y escritos, los bits que tiene el encabezado R son aquellos que solo pueden ser leídos.
 1. **SPEN:** bit que activa el puerto serial.
 0: puerto serial desactivado.
 1: puerto serial activado es decir configura RC7/RX/DT y RC6/TX/CK como los pines del puerto serial.
 2. **RX9:** bit que activa el 9 bit de recepción.
 0: 8 bits de recepción.

1: 9 bit de recepción.

3. SREN: bit que activa solo recepción, solo se tiene en cuenta en el modo síncrono, este bit se pone en 0 cuando se completa la recepción.

0: desactiva solo recibir.

1: activa solo recibir.

4. CREN: bit que activa la recepción continúa.

Modo asíncrono

0: desactiva la recepción continua.

1: activa la recepción continua.

Modo síncrono:

0: desactiva la recepción continua.

1: activa la recepción continua y hasta que el bit CREN sea 0.

5. ADDEN: bit que activa la detección de dirección.

0: desactiva la detección de dirección todos los bits se reciben y el bit 9 se puede utilizar como bit de paridad.

1: activa la detección de dirección y permite interrumpir la carga del buffer.

6. FERR: bit de framing de errores de bits.

0: no hay framing de error

1: framing de error

7. OERR: bit de error de superación.

0: no hay error de rebasamiento.

1: hay error de superación, este bit se borra al borrar el CREN.

8. RX9D: bit 9 de la recepción de datos, se puede usar como bit de paridad.

2.6.2.2 Manejo del módulo USART del PIC 16F877a

En el dispositivo de supervisión se implementaron una serie de algoritmos en los que se utiliza la comunicación serial asíncrona entre el PIC 16F877a y el modem Bluetooth - BlueSMiRF RP- SMA, para realizar esta comunicación que en la mayoría de los casos es bidireccional se utilizaron las funciones tomadas de la directiva *rs232* que se activa en el programa del PIC para el envío y la recepción de los datos.

La función usada para la transmisión de los datos desde el PIC 16F877a hacia el modem Bluetooth - BlueSMiRF RP- SMA es *Putc(A)*; en esta función A es el carácter ASCII que se desea transmitir. El dato ASCII se envía a través del pin xmit que en este caso el pin RC6/TX/CK del PIC.

La función usada para la recepción de los datos que se envían desde el modem Bluetooth - BlueSMiRF RP- SMA hacia el PIC16F877a es *D = getch()*; donde D es la variable en donde se almacena el valor ASCII del dato recibido. El dato se recibe en el PIC a través del pin rcv que para el caso particular es el pin RC7/RX/DT.

En la tabla 9 se muestra como se configura el modulo USART en lenguaje C y como se llama a las funciones.

Tabla 9. Configuración del USART en C.

```
#use delay(clock=4000000) //configuracion del reloj del PIC
#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7) // configuración del USART
////////funciones de USART////////
D = getch(); // recepción de datos ASCII
Putc(A); // transmisión de datos ASCII
```

2.7 MÓDULO MULTIPLEXORES ANALÓGICOS

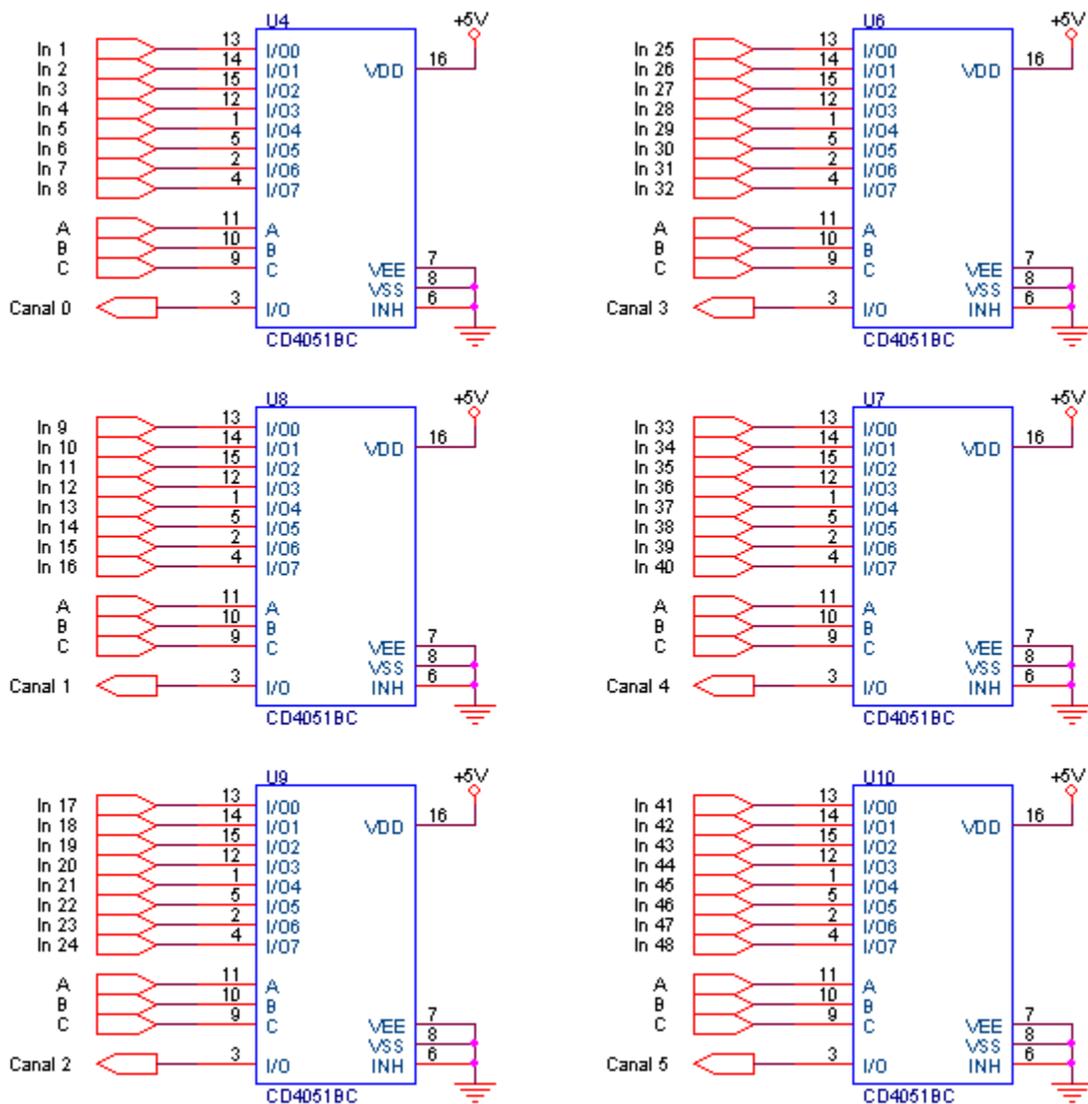
El dispositivo de supervisión tiene la capacidad de supervisar 24 ratas (48 señales) utilizando un PIC 16F877a que cuenta con 8 canales en su módulo de conversión análogo a digital, para lograr esta expansión en la capacidad de

canales de entrada para el PIC se recurrió al uso de multiplexores analógicos CD4051.

El CD4051 es un multiplexores analógicos de 8 entradas, para lograr las 48 entradas que se utiliza el dispositivo de supervisión se usaron 6 multiplexores CD4051 y 6 canales de modulo A/D del PIC.

2.7.1 Descripción de los pines del CD4051

Figura 26. Diagrama circuital de los multiplexores analógicos CD4051



Tomado como referencia la figura 26 se describen los pines del CD4051:

- **VDD:** es el pin de la alimentación positiva o +5VDC
- **VSS:** este pin se conecta a la tierra de alimentación.
- **VNH:** este pin tiene como función inhibir las entradas cuando se encuentra en un valor lógico alto y de no hacer nada cuando esta en un valor lógico bajo. En el dispositivo de supervisión este pin está conectado a la tierra de alimentación.
- **I/O0 a I/O7:** estos pines corresponden a las entradas o salidas de CD4051 dado que también se puede configurar como demultiplexor. En el dispositivo estos pines se conectan a las señales a supervisar.
- **I/O:** es el pin que se usa como salida del multiplexor o como entrada cuando se trabaja el CD como demultiplexor. En el dispositivo este pin se conecta a los canales modulo A/D del PIC 16F877a en este orden que se muestra en la figura 26.
- **A, B, C:** estos pines son los pines de control de la multiplexión en donde A funciona es el bit menos significativo y C el bit más significativo. En el dispositivo de supervisión están conectados así, los pines A, B, C con los pines RC0, RC1, RC2 del PIC 16F877a respectivamente.

En la tabla 10 se muestra la tabla de verdad del multiplexor CD 4051, la cual se tuvo en cuenta para crear la estrategia de control del PIC 16F877a

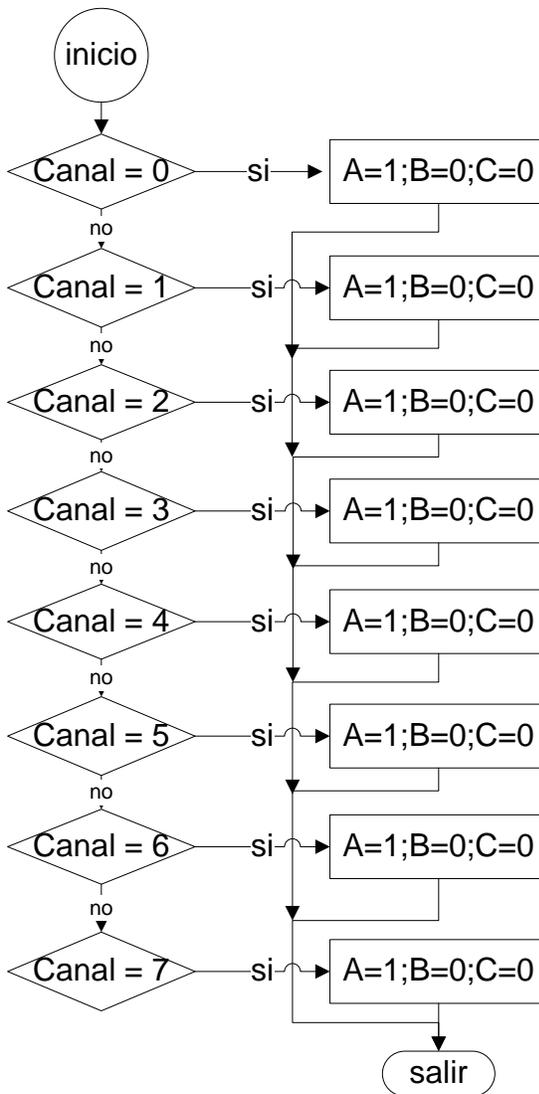
Tabla 10. Tabla de verdad del CD 4051

Estado de las entradas				I/O activa
C	B	A	INH	
0	0	0	0	0
0	0	1	0	1
0	1	0	0	2
0	1	1	0	3
1	0	0	0	4
1	0	1	0	5
1	1	0	0	6
1	1	1	0	7
...	1	...

2.7.2 Manejo de los multiplexores CD4051

El control de los multiplexores CD4051 del dispositivo de supervisión se realiza a través del PIC16F877a en el que se programo una rutina para tal fin.

Figura 27. Diagrama de flujo de la rutina de manejo de los CD4051.



En el diagrama de flujo de la figura 27 se muestra como se elige la entrada de cada multiplexor ubicando en las entradas de control del mismo la combinación binaria adecuada para cada habilitar la entrada que se desee del multiplexor.

Por otro lado la rutina de la tabla 11 es el algoritmo en lenguaje C programado en el PIC16F877a para hacer el control de los multiplexores CD4051.

Tabla 11. Rutina en C para el control de los multiplexores CD4051

```
//////////////////////////////////// rutina control de los multiplexores CD4051 //////////////////////////////////////
void multiplexar(dato){
switch (dato){          //limites...
case 0:  output_low(pin_c0);output_low(pin_c1);output_low(pin_c2);break;
case 1:  output_high(pin_c0);output_low(pin_c1);output_low(pin_c2);break;
case 2:  output_low(pin_c0);output_high(pin_c1);output_low(pin_c2);break;
case 3:  output_high(pin_c0);output_high(pin_c1);output_low(pin_c2);break;
case 4:  output_low(pin_c0);output_low(pin_c1);output_high(pin_c2);break;
case 5:  output_high(pin_c0);output_low(pin_c1);output_high(pin_c2);break;
case 6:  output_low(pin_c0);output_high(pin_c1);output_high(pin_c2);break;
case 7:  output_high(pin_c0);output_high(pin_c1);output_high(pin_c2);break;
}
}
} //-----fin función-----
```

2.8 MODULO A/D²⁴

El PIC 16F877a es un microcontrolador de la familia 16F8XX de la empresa microchip con múltiples funciones, una de estas es la de su modulo de conversor análogo a digital A/D de 8 o 10 bits. El modulo A/D cuenta con cuatro registros distribuidos así:

- **ADRESH:** Es el registro que almacena la parte alta o más significativa de la conversión.
- **ADRESL:** Es el registro encargado de almacenar la parte menos significativa de la conversión o en su defecto la parte baja.
- **ADCON0:** Es el registro que controla en funcionamiento del conversor, es llamado también registro de control 0.

²⁴ <http://ww1.microchip.com/downloads/en/devicedoc/21754E.pdf>

- **ADCON1:** Llamado registro de Control 1, es el encargado de la configuración de los pines del puerto del modulo A/D.

2.8.1 Registros módulo A/D del PIC 16F877a

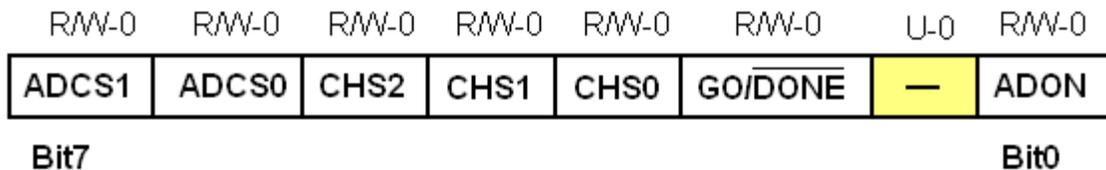
El conversor A/D del PIC 16F877a cuenta con 8 canales de entrada analógicos y se puede configurar para trabajar con 8 o 10 bits por conversión, esta configuración se logra a través de la manipulación del los registros ADCON0 y ADCON1.

Para entender la forma en que fue utilizado en módulo A/D en la construcción del dispositivo de supervisión es necesario entender el funcionamiento de los registros de control, cada registro de control está formado por 8 bits cada uno.

Tomando como referencia la figura 29 y la figura 30 se explicaran cada uno de los bits que componen el registro de control ADCON0 y ADCON1 respectivamente.

- **ADCON0** es el registro que controla el funcionamiento del conversor A/D.

Figura 28. Estructura del registro ADCON0



1. **ADCS1 y ADCS0:** son los bits con los que se selecciona el reloj de conversión del módulo A/D.

00 = FOSC/2.

01 = FOSC/8.

10 = FOSC/32.

11 FRC (reloj derivado de un circuito oscilatorio RC).

2. **CHS2 a CHS0:** son los bits que se encargan de seleccionar el canal analógico del módulo A/D.

000 = canal 0, (RA0/AN0).

001 = canal 1, (RA1/AN1).

010 = canal 2, (RA2/AN2).

011 = canal 3, (RA3/AN3).

- 100 = canal 4, (RA5/AN4).
- 101 = canal 5, (RE0/AN5).
- 110 = canal 6, (RE1/AN6).
- 111 = canal 7, (RE2/AN7).

3. GO/DONE: es el bit de estado de la conversión A/D.

Si el bit ADON = 1.

GO/DONE = 1, la conversión A/D está en proceso.

GO/DONE = 0, la conversión A/D ha terminado, este bit se borra por hardware automáticamente.

4. ADON: es el bit que activa o desactiva el conversor A/D.

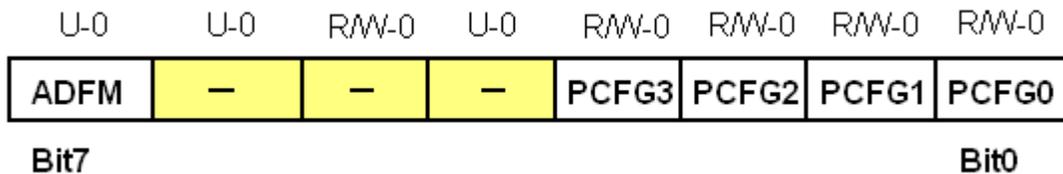
ADON = 1, el conversor está activado encendido.

ADON = 0, el conversor está desactivado o apagado.

5. El bit 1 no se utiliza y se asume con un valor = 0, por otro lado los demás bits pueden ser leídos o escritos por programación.

- **ADCON1** es el registro de control encargado de configurar los pines del puerto del modulo A/D del PIC 16F877a.

Figura 29. Estructura del registro ADCON1



- 1. EI ADFM:** A/D: es el bit que selecciona el formato de resultado de la conversión A/D.
 1 = justifica a la derecha, es decir que los 6 bits más significativos del ADRESH son leídos como 0.
 0 = justifica a la izquierda, es decir que los 6 bits menos significativos del ADRESL son leídos como 0.
- 2. PCFG3 a PCFG0:** son los bits que controlan la configuración del puerto. En la tabla 2 se muestran las posibles combinaciones de estos bits y sus respectivas configuraciones.
- 3.** Los bits 5 a 4 no se utiliza y se asume con un valor = 0, por otro lado los demás bits pueden ser leídos o escritos por programación.

Tabla 12. Posibles configuraciones de los canales del módulo A/D del PIC 16F877a

PCFG3: PCFG0	AN7 RE2	AN6 RE1	AN5 RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF +	VREF -	CHAN/ Refs ⁽¹⁾
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
0110	D	D	D	D	D	D	D	D	VDD	VSS	0/0
0111	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2
A = Entrada analógica, D = Entrada /salida digital											
1: En esta columna se indica el número de canales de entrada habilitados como analógicos y el número de canales usados como voltajes de referencia											

2.8.2 Manejo del módulo A/D

El uso del módulo A/D de PIC 16F877a resultó muy conveniente para el diseño en implementación del dispositivo de supervisión de la prueba de consumo de sacarosa en ratas, ya que a través de un mismo dispositivo en este caso el PIC se pudo tanto efectuar las siguientes funciones:

- Conversión análoga a digital usando 8 bits, de las señales de la prueba.
- Contralar los un grupo de multiplexores analógicos CD4051, cuya función es la de incrementar los canales de conversión.

- Almacenar los datos representativos de la conversión en una memoria EEPROM 24LC512 externa a través de el protocolo I2C si se trabaja el dispositivo en modo directo.
- Enviar los datos vía protocolo RS232 al Modem Bluetooth - BlueSMiRF RP-SMA quien a su vez los envía al PC o la PocketPC usando la especificación Bluetooth.

En la tabla 13 se muestran las rutinas que fueron programadas en el PIC16F877a para que realizar las acciones que se describieron anteriormente. Por otra parte en la figura 30 se presenta el diagrama de flujo correspondiente al algoritmo programado en el PIC para la adquisición de datos del dispositivo.

Figura 30. Diagrama de flujo de las adquisición de señales del dispositivo de supervisión

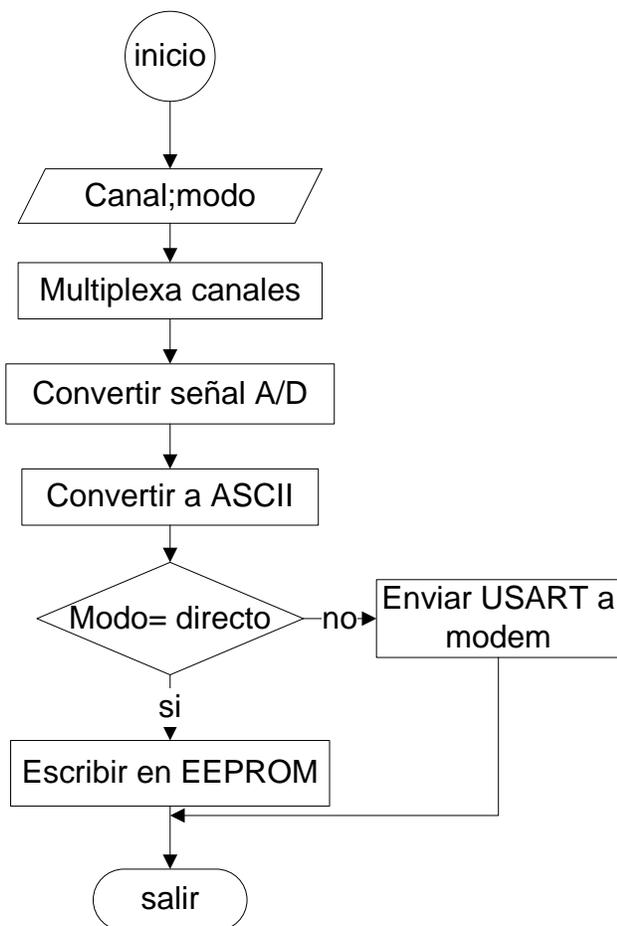


Tabla 13. Código en C de la adquisición A/D con el PIC16F877a.

```

////////////////////////////////// inicialización del modulo A/D del PIC16f877a //////////////////////////////////
#include <16f877A.h>

#device adc=8

#fuses XT,NOWDT,NOPROTECT,NOLVP,PUT,BROWNOUT

#use fast_io(b)

#use delay(clock=4000000)

#include <adc.h>

#include <input.c>

////////////////////////////////// rutina de conversión A/D //////////////////////////////////

void adquirir(){

multiplexar (); // rutina que controla los multiplexores analógicos

setup_adc(ADC_CLOCK_INTERNAL); //configuración del reloj del conversor A/D

set_adc_channel(canal); // selección del canal del conversor A/D

val = read_adc(); //lectura del valor de la conversión de la señal

NUM_ASCII(val); //rutina que transforma las unidades, decenas y centenas del
//valor adquirido en ASCII

putc(centenas); //envió de centenas del valor de la señal usando rs232

putc(decenas); //envió de decenas del valor de la señal usando rs232

putc(unidades); // envió de unidades del valor de la señal usando rs232

contador++;
conteo++;
conteo1++;

if(temp3==1)

```

```

        {
            goto sin;
        }
if(temp3==2)
    {
        goto sin;
    }

if(num2>=conteo1)
    {
        write_ext_eeprom(temp1, centenas); // escritura en memoria 24LC512
        temp1++;
        write_ext_eeprom(temp1, decenas); //escritura en memoria 24LC512
        temp1++;
        write_ext_eeprom(temp1, unidades); //escritura en memoria 24LC512
    }

sin:
if(contador==8)
    {
        contador=0;
        canal++;
    }

if(temp==1)
    {
        temp=0;
    }
else
    {
        temp=1;
    }

delay_ms(1);

if(temp3==2)
    {
        delay_ms(1);
    }

}//-----fin función-----

```

3. LABVIEW 7.1

LabVIEW (*Laboratory Virtual Instrument Engineering Workbench*) es un sistema de desarrollo creado por National Instruments que utiliza un lenguaje de programación gráfica que es conocido como lenguaje G, en LabVIEW se puede generar aplicaciones para la instrumentación, control de dispositivos, la supervisión de procesos, y la adquisición y procesamiento digital de señales, entre otras.

Una de las principales ventajas que ofrece LabVIEW es la versatilidad que posee para crear interfaces de usuario amigables, así como capacidad de comunicación con dispositivos e instrumentos que manejan estándares de software y comunicación como COM, OLE, DDE, DLL o ActiveX para el caso de Windows.

3.1 INSTRUMENTOS VIRTUALES (VI'S) EN LABVIEW

Las aplicaciones que se desarrollan en el ambiente de LabVIEW reciben el nombre de instrumentos virtuales o VI's, para el desarrollo de un VI el usuario cuenta con dos partes una es el panel frontal que es donde se desarrolla la interfaz grafica de usuario y la otra es diagrama de bloques que es el lugar donde se desarrolla la programación gráfica que hará funcionar el VI.

En el panel frontal el programador cuenta con una paleta de controles que como su nombre lo indica contiene los controles e indicadores que se requieren para el desarrollo de una interfaz de usuario amigable, en el diagrama de bloques se cuenta con una paleta de funciones con las que se programan las estrategias del VI.

3.1.1 Sub VI'S

Los subVI's son la analogía de las subrutinas de los lenguajes de programación convencional, por este motivo los subVI's son VI simples o básicos que se desarrollan para ser llamadas desde VI más complejos en los que se requiere un espacio gráfico considerable para desarrollar la programación. En el desarrollo de las interfaces creadas en LabVIEW para la supervisión de la prueba de consumo de sacarosa en ratas se utilizaron varios subVI's para hacer más eficiente el instrumento virtual principal de la interfaz.

3.2 Comunicación entre el modem Bluetooth – BlueSMiRF RP- SMA y la aplicación de LabVIEW.

Como se ha venido mencionado el dispositivo de supervisión tiene un modo de manejo en el que se puede controlar a través de un PC y otro en el que se puede controlar a través de una PocketPC. En ambos modos el dispositivo de supervisión se comunica con las aplicaciones (VI's) creadas en LabVIEW usando el modem Bluetooth – BlueSMiRF RP- SMA y ciertas funciones de propias de LabVIEW que permiten la comunicación mediante le especificación Bluetooth.

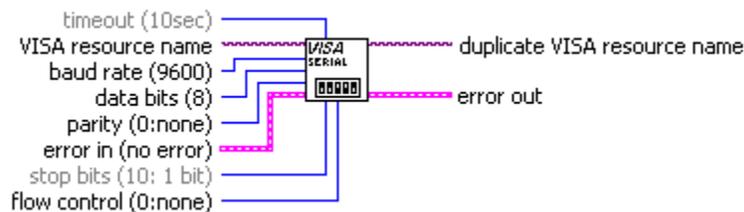
A continuación se describen estas funciones tanto para la VI del PC como para el VI de la PocketPC que permiten la lectura y la escritura de datos desde y hacia el modem Bluetooth.

3.2.1 Funciones para la lectura y escritura de datos en el PC

En la figura 31 se muestra la función que se utiliza el VI para la configuración del puerto COM al cual el asistente de Windows para la detección y configuración de dispositivos Bluetooth le asigno al modem Bluetooth, esta función es quien configura las características de la comunicación.

Descripción de las entradas y salidas de la función:

Figura 31. Función de configuración de la comunicación



- **Timeout:** es donde se define el tiempo en máximo que espera el VI para generar un mensaje de error si no ha podido leer o enviar un dato.
- **Puerto I/O:** es donde se asigna el COM que asignó el asistente de Windows después de la detección del modem, normalmente el COM4.
- **Baud rate:** en esta parte se define la rata de baudios a usara en la transmisión y recepción de los datos, para el caso del dispositivo se asigno 9600.

- **Data bits:** este parámetro define el número de bits que se recibirá o enviará el VI por cada dato, para el caso de la aplicación que se asignó 8 bits o un byte.
- **Parity:** define la paridad de la comunicación, para el caso del dispositivo se definió sin paridad.
- **Stop bit:** aquí se define el número de bit que terminan el frame se pueden utilizar 1, 1.5 o 2 bits de parada, para el caso del VI del dispositivo se utilizó 1 bit de parada.
- **Flow control:** es donde se controla en control de flujo de datos de la comunicación, en el VI del dispositivo de supervisión no se utiliza el control de flujo.
- **Duplicate VISA resource name:** es simplemente la salida en donde se duplica el nombre de los recursos que se utilizan para la comunicación.
- **Error out:** es la parte de la función que se encarga de mostrar los errores que se produzcan en la comunicación.

3.2.1.1 Función de lectura de datos de LabVIEW para el PC

El VI implementado utiliza la función de lectura de datos tanto cuando supervisa la prueba como cuando lee los datos almacenados en la memoria EEPROM 24lc512 del dispositivo.

La función que se usó para la lectura de los datos por parte del PC a través del VI implementado fue la que se muestra en la figura 32. En esta función se notan varias entradas y salidas que se describen a continuación.

Figura 32. Función de lectura de datos de LabVIEW para el PC



- **ViSA resource name:** es donde se ingresa la configuración de puerto que se utilizará para la comunicación, en el caso del VI implementado esta entrada está cableada a la salida de la función de configuración que se describió en el apartado anterior.
- **Byte count:** aquí se define el número de bits que la función leerá cuando se está utilizando, para el caso del dispositivo este valor depende de que

información se esté leyendo, por ejemplo para el consumo se deben leer 3 bytes pero para la fecha se deben leer 6 bytes.

- **Error in:** aquí se definen las condiciones necesarias para que se produzca un error, en el caso del dispositivo se dejó en su estado por defecto que es sin error.
- **Dup VISA resource name:** es una salida en donde se muestra el nombre del recurso que la comunicación utiliza, en el VI estas salidas se cablean a las entradas VISA resource name de las funciones contiguas.
- **read bufer:** es la salida en donde se escribe el dato que se leyó, en este caso lo que se envía del dispositivo.
- **Error out:** es la parte de la función que se encarga de mostrar los errores que se produzcan en la comunicación.

3.2.1.2 Función de escritura de datos de LabVIEW para el PC

Al igual que la función de lectura, la de escritura se utiliza tanto para supervisar como para leer los datos almacenados en la memoria del dispositivo, cabe resaltar que en el VI se utiliza para la sincronización con el dispositivo.

En la figura 33 se muestra la función de escritura y a continuación se describirán sus entradas y salidas.

Figura 33. Función de escritura de datos de LabVIEW para el PC



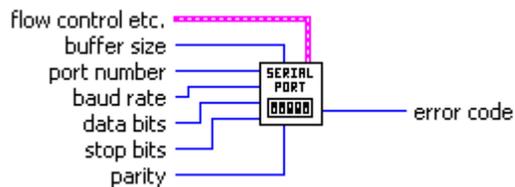
- **ViSA resource name:** es donde se ingresa la configuración de puerto que se utilizará para la comunicación, en el caso del VI implementado esta entrada esta cableada a la salida de la función de configuración o de lectura que la preceda.
- **Write buffer:** es la entrada en donde se deben ubicar el dato que se desea escribir en el puerto para ser enviado.
- **Error in:** aquí se definen las condiciones necesarias para que se produzca un error, en el caso del dispositivo se dejó en su estado por defecto que es sin error.

- **Dup VISA resource name:** es una salida en donde se muestra el nombre del recurso que la comunicación utiliza, en el VI estas salidas se cablean a las entradas VISA resource name de las funciones contiguas.
- **Return count:** retorna el valor del número de Bytes que se escribieron en el puerto.
- **Error out:** es la parte de la función que se encarga de mostrar los errores que se produzcan en la comunicación.

3.2.2 Funciones para la lectura y la escritura de datos en la PocketPC

Al igual que en el VI desarrollado para el PC, el que se implementó para la PocketPC requiere del uso de funciones que escriban y lean datos desde y hacia el modem Bluetooth – BlueSMiRF RP- SMA, pero antes de realizar cualquier tipo de envío o recepción de datos se necesita tener configurada la comunicación. En la figura 34 se muestra la función que permite la configuración de la comunicación entre el modem y la aplicación de LabVIEW de la PocketPC

Figura 34. Función de configuración del puerto



A continuación se describen las entradas y las salidas de la función:

- **Flow control etc:** esta entrada de la función permite la configuración del control de flujo de la comunicación, para el caso del VI implementado para la supervisión de la prueba este parámetro se deja en su estado por defecto es decir sin control de flujo.
- **Buffer size:** define el tamaño del buffer del puerto por donde se enviaran los datos para el caso del VI desarrollado se utilizó 1024 bits.
- **Port number:** es el número de puerto que se le asigna al modem Bluetooth por parte de la PocketPC.
- **Baud rate:** se define la tasa de baudios en la que se realizara la comunicación, para el caso particular se utilizara 9600 al igual que en el PC.

- **Data bits:** en este pin se define el número de bits que conforman un dato, en este caso se configuro a 8 bits es decir 1 bytes por cada dato.
- **Stop bit:** en esta entrada se define el número de bits que se utilizan para terminar la comunicación.
- **Parity:** define la paridad de la comunicación, para el caso del VI desarrollado para la PocketPC no se utiliza paridad.

3.2.2.1 Función de lectura de datos del LabVIEW para la PocketPC

La función de lectura de LabVIEW usada en la PocketPC se muestra en la figura 35, esta función se utiliza para leer constantemente los datos provenientes del dispositivo de supervisión. A continuación se describen las entradas y salidas de la función:

- **Port number:** es la entrada donde se debe ingresar el número de puerto que se usara para la comunicación, el cual es asignado por la PocketPC.
- **Requested byte count:** donde se indica a la función cuantos bytes debe leer.
- **String read:** es la salida donde aparece el dato leído.
- **Error code:** indica cuando se produce un error en la lectura de un dato.

Figura 35. Función de lectura en LabVIEW usada para la PocketPC



3.2.2.2 Función de escritura de datos de LabVIEW para la PocketPC

La función escritura de datos se utilizo para el envió hacia el modem del dispositivo de supervisión de los datos de sincronización para la comunicación. En la figura 36 se muestra la función de escritura de LabVIEW usada para la el VI desarrollado para la PocketPC, a continuación se describen las entradas y salidas de la misma:

- **Port number:** es la entrada donde se debe ingresar el número de puerto que se usará para la comunicación, el cual es asignado por la PocketPC.
- **String to write:** es la entrada donde se ubica el dato que se desea escribir en el puerto.
- **Error code:** indica cuando se produce un error en la lectura de un dato.

Figura 36. Función de escritura de LabVIEW usada para la PocketPC



4. FUNCIONAMIENTO

En este capítulo se explica el funcionamiento del dispositivo de supervisión de la prueba de consumo de sacarosa en ratas. El dispositivo cuenta con tres formas de manejo las cuales pueden ser elegidas por el usuario que está trabajando con el dispositivo. Estas formas de manejo son en forma directa, usando un PC y usando una PocketPC.

Después de encender el dispositivo aparece un mensaje de bienvenida para el usuario que se muestra en la figura 37, este mensaje dura 3 segundos para dar paso después al mensaje del menú principal de la figura 38 en donde el usuario puede elegir el modo de manejo que desee usar.

Figura 37. Mensaje de bienvenida para el usuario



Figura 38. Mensaje del menú principal del dispositivo



4.1 MODO DIRECTO

El modo directo se desarrolló con el propósito de proporcionar movilidad al dispositivo de supervisión, es decir que el usuario puede aplicar un número amplio de pruebas sin necesidad del uso de un computador. En el modo de manejo directo el dispositivo tiene la capacidad de almacenar un buen número de pruebas dado que cuenta con una memoria EEPROM 24LC512, la cual permite el almacenamiento de 512 kbits de información.

4.1.1 Supervisión en el modo directo.

Para la supervisión de la prueba de consumo de sacarosa en ratas mediante el uso del dispositivo en modo directo, el usuario simplemente debe seguir los menús que se van presentado en el LCD 2X16 e ingresar la información que el dispositivo le van pidiendo a través del teclado matricial 4X4.

A continuación se explican los pasos que se deben seguir para supervisar una prueba de consumo de sacarosa.

- Después de haber seleccionado el modo directo en el menú principal (figura 38) al usuario se le presentara un mensaje de confirmación para el ingreso de los datos de la figura 39, si el usuario presiona la letra C en el teclado matricial 4X4 se continuará al siguiente menú pero si por el contrario presiona la letra D se retorna al menú principal.

Figura 39. Mensaje de confirmación del ingreso de los datos



- Si en usuario presiono la letra C se desplegará el mensaje que se muestra en la figura 40, en este menú el usuario debe elegir la letra que corresponde al valor del porcentaje de concentración de sacarosa que contendrá la solución con la que se realizara la prueba.

Figura 40. Mensaje para elegir el porcentaje de sacarosa que se utilizará en la prueba.



- Después del usuario elija el porcentaje de concentración, se desplegará el mensaje que se muestra en la figura 41 en el cual se pide la confirmación del porcentaje de concentración que se eligió, el usuario puede confirmar si selecciona la opción *Segir* o retroceder al menú anterior si elije la opción *Atrás*.

Figura 41. Mensaje de confirmación de porcentaje de concentración



- Después de confirmar el porcentaje de concentración de sacarosa, se desplegará el mensaje de la figura 42, en este menú el usuario tiene la posibilidad de elegir tres tiempos de duración de la supervisión de la prueba.

Figura 42. Mensaje de elección de tiempos de duración



- Una vez el usuario elija el tiempo de duración de la prueba, aparecerá el mensaje de confirmación de la figura 43, en este punto el usuario debe confirmar si el tiempo elegido es el correcto presionado la tecla correspondiente a *Seguir* o regresando al menú eligiendo la opción *Atrás*.

Figura 43. Mensaje de confirmación del tiempo de duración.



- El mensaje de la figura 44 se muestra después de que usuario confirme la duración de la prueba, en este mensaje el usuario debe ingresar el número de ratas a las que se les supervisará la prueba, el número debe escribirse como se indica en la cabecera del mensaje, como por ejemplo si se desea supervisar 4 ratas se debe escribir 03. Si el usuario escribe un número mayor a 24 o menor a 01 se mostrará el mensaje de error de la figura 45 y si el usuario presiona una letra se mostrara el mensaje de la figura 46, en ambos casos retorna al menú de la figura 38 después de 3 segundos.

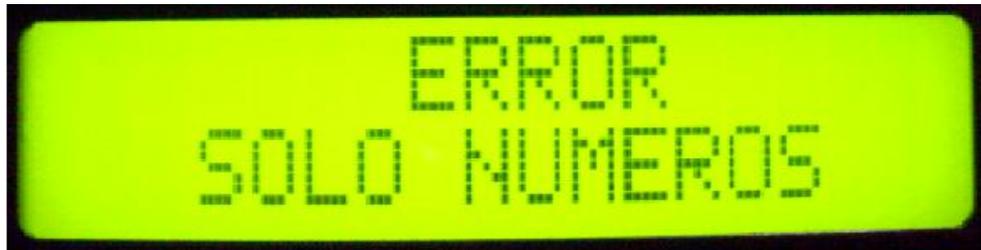
Figura 44. Mensaje para el ingreso del número de ratas



Figura 45. Mensaje de error de números



Figura 46. Mensaje de error de ingreso de letras



- Después del ingreso del número de ratas se muestra el mensaje de confirmación de la figura 47, en el que el usuario puede elegir *Segir* para confirmar o *Atrás* para regresar al menú anterior.

Figura 47. Mensaje de confirmación del número de ratas



- Si el usuario confirma el número de ratas, se desplegará el mensaje de la figura 48 por 4 segundos y en el que se indica al usuario que debe ingresar el ID de cada una de las ratas y que este no debe ser de más de 4 caracteres.

Figura 48. Mensaje de indicaciones para el ingreso de los ID de las ratas



- En el mensaje de la figura 49 se pide al usuario que ingrese el ID de 4 caracteres de cada rata. Después de ingresado el ID por el usuario aparece el mensaje de la figura 50 en el que el usuario debe confirmar si el ID que

ingreso es correcto eligiendo la opción *Segir* o regresando al menú anterior si presiona *Atrás*.

Figura 49. Mensaje de ingreso del ID de la rata



Figura 50. Confirmación del ID de la rata ingresado



- Después del usuario confirmar el ultimo ID del número de ratas a las que les va a supervisar la prueba se inicia la adquisición de los datos y se muestra en la LCD el mensaje de la figura 51, hasta que se termine la adquisición.

Figura 51. Mensaje de adquisición en curso



- Después de terminada la adquisición se muestra el menú de la figura 52 en el que se le indica al usuario que la prueba ha finalizado y que si presiona a

la letra A puede retornar al menú principal, o en su defecto el usuario puede apagar el dispositivo.

Figura 52. Mensaje de fin de la prueba



4.2 MODO PC

En el modo PC el usuario puede controlar el dispositivo de supervisión desde un PC para supervisar una o varias pruebas con una máximo de 24 ratas por prueba y almacenar los datos que se arrojan de cada una de ellas en un en uno o varios archivos planos. En este modo también se tiene la posibilidad de descargar los datos almacenados en la memoria EEPROM 24LC512 del dispositivo y guardarlos en un archivo plano en el PC.

Para acceder a alguno de los dos modos de manejo con los que el dispositivo cuenta en el modo PC el usuario debe elegir *PC* en el menú principal. Después de ingresar al modo PC se mostrará el mensaje de la figura 53.

En este menú se puede elegir los dos modos de manejo con los que cuenta el PC uno es el de supervisar la prueba con el PC el cual se elije seleccionado *Adquirir PC* y si elije la opción *Descargar PC* se ingresa al modo de leer los datos de la memoria EEPROM del dispositivo a través de PC.

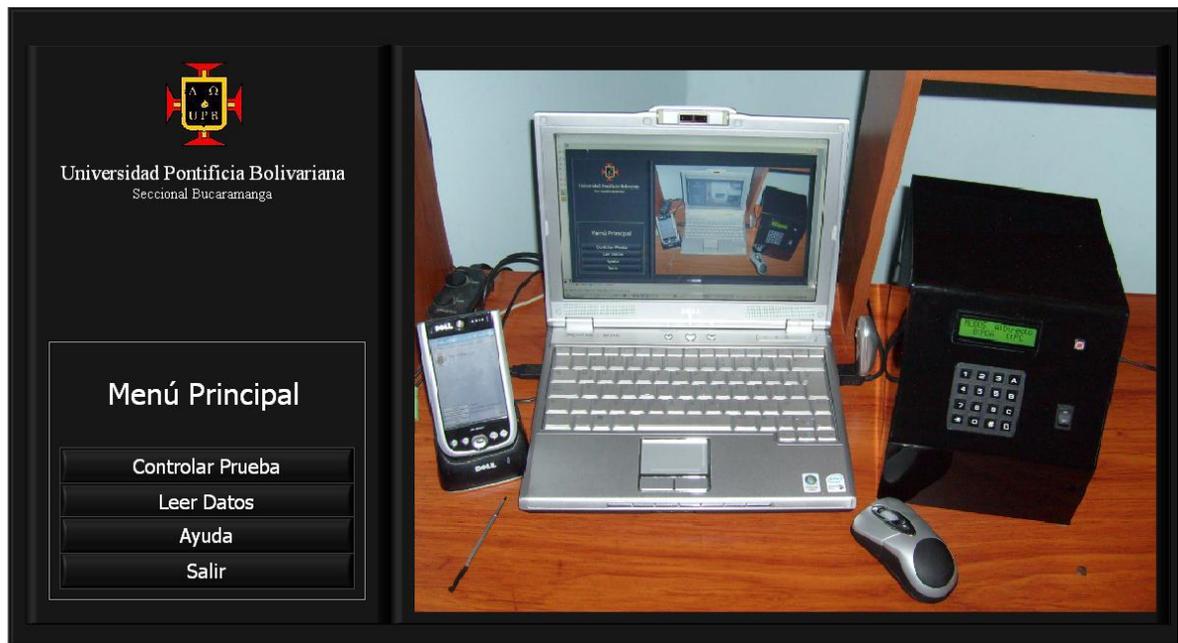
Figura 53. Menú del modo PC



La interfaz de usuario desarrollada en LabVIEW para el control de dispositivo desde el PC que se muestra en la figura 54 está formada por cuatro botones relacionados de la siguiente manera:

- **Controlar la prueba:** accede al modo de control del dispositivo a través del PC.
- **Leer datos:** se accede al menú correspondiente a la descarga de los datos almacenados en la memoria EEPROM del dispositivo hacia el PC.
- **Ayuda:** al pulsar este botón el usuario accede a la ayuda con la que cuenta la aplicación creada en LabVIEW.
- **Salir:** como su nombre lo indica cierra la aplicación.

Figura 54. Interfaz gráfica de usuario del modo PC



4.2.1 Supervisión en modo PC

Inicialmente para usar el modo PC del dispositivo se debe detectar el modem Bluetooth – BlueSMiRF RP- SMA en el PC mediante el asistente de Windows para detección de dispositivos Bluetooth, después de realizado esto se debe elegir en el dispositivo en el menú PC del mismo (ver figura 53) la opción *Adquirir PC*. Después aparecerá el mensaje de la figura 55 que invita al usuario a confirmar la

supervisión a través del PC eligiendo la *opción Iniciar Adqui* o regresar al menú principal si se elige la opción con *Menu Principal*.

Figura 55. Mensaje de confirmación de la supervisión a través del PC



Después de confirmada al supervisión de la prueba usando el PC se mostrará en el dispositivo el mensaje de la figura 56 en el que se invita el usuario a pulsar el botón de *Iniciar prueba* en la interfaz de LabVIEW en el PC.

Figura 56. Mensaje de invitación a iniciar la prueba desde el PC



Después de que el usuario haya configurado el dispositivo para supervisar la prueba a través del modo PC, se debe configurar el PC para iniciar la supervisión y esto se logra a través de la interfaz de LabVIEW que se muestra en la figura 57.

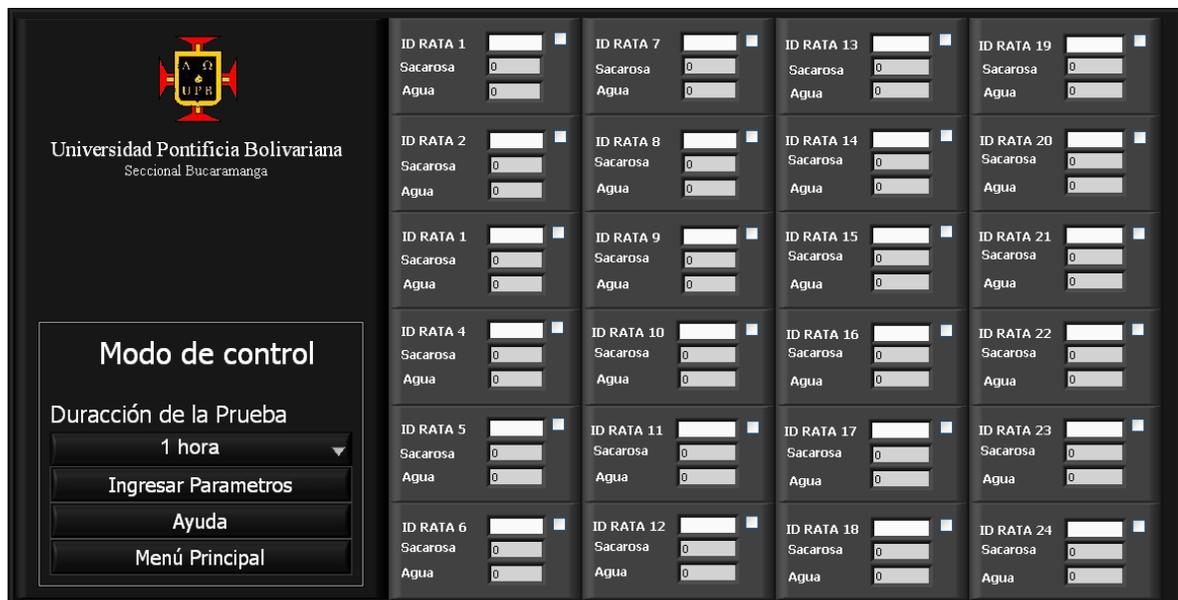
En la interfaz del LabVIEW el usuario debe elegir la duración de la prueba, elegir las ratas a las cuales se les supervisara la prueba, ingresar los ID de cada una de las mismas, el porcentaje de concentración de la sacarosa que se va a utilizar.

Para poder iniciar la prueba se deben seguir los siguientes pasos:

- El usuario debe seleccionar la duración de la prueba en el menú desplegable.
- Después el usuario debe pulsar el botón *Ingresar Parámetros*, automáticamente se desplegará una ventana en la que el usuario debe ingresar el porcentaje de concentración de sacarosa que se utilizara en la prueba.

- El usuario debe seleccionar las ratas a las que les desea aplicar la prueba y debe ingresar su ID después debe presionar el botón siguiente o atrás si desea modificar la duración o porcentaje de sacarosa de la prueba.
- Después debe presionar el botón *Iniciar prueba* y la supervisión se inicia.
- Al terminara la prueba se desplegará una ventana de dialogo en el que el usuario debe ingresar el nombre, la ubicación y la extensión del archivo donde quiere que se genere el archivo de reporte, es recomendable que la extensión que se utilice sea .XLS para de esta manera generar una tabla bien relacionada (ver figura 59)
- Para salir, después de terminada la prueba se debe presionar el botón *Menu Principal*.

Figura 57. Interfaz de control de la supervisión a través del PC



En la figura 58 se muestra el diagrama de transición de estados de la aplicación creada en LabVIEW para la supervisión de la prueba de consumo de sacarosa en ratas usando una interfaz grafica de usuario en el PC.

Figura 58. Diagrama de transición de estados del modo supervisar con PC

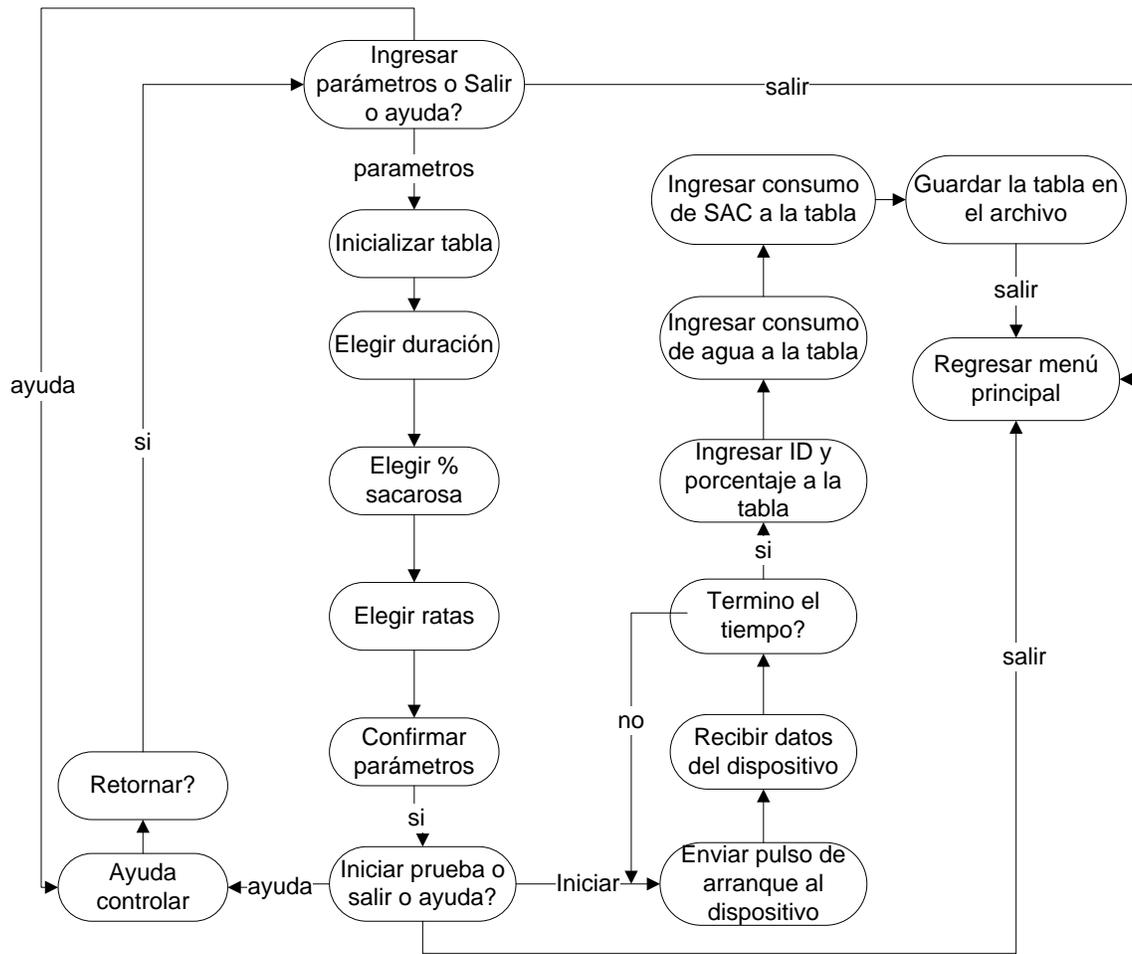


Figura 59. Tabla de resultados obtenidos de la aplicación de una prueba

PRUEBA DE CONSUMO DE SACAROSA EN RATAS

miércoles, 21 de mayo de 2008, 13:38:24

En la siguiente tabla se muestran los resultados de la prueba de consumo de sacarosa en ratas aplicada en la fecha indicada.

ID de la rata	% de sacarosa	Consumo de sacarosa	Consumo de agua
	16	61	70
	16	76	75
	16	79	78
	16	84	83
	16	113	115
	16	121	117
	16	124	120
	16	124	118

Universidad Pontificia Bolivariana
Seccional Bucaramanga

4.2.2 Lectura de datos en modo PC

El dispositivo de supervisión cuenta con una memoria EEPROM 24LC512 que permite el almacenamiento de los datos relacionados con la prueba (ID rata, consumo de agua, consumo de sacarosa, fecha y hora) cuando el dispositivo se trabaja en modo directo. En el modo PC (ver figura 54) se cuenta con un modo de lectura el cual permite la descarga de los datos que están almacenados en la memoria EEPROM del módulo.

Después de acceder al modo de lectura de datos en el menú del modo PC (figura 53) eligiendo la opción *Descargar PC*, el usuario se encontrará con el mensaje de confirmación que se muestra en la figura 60 que le presenta el módulo, en este punto si el usuario selecciona *Iniciar Down* el dispositivo pasara al estado de espera para iniciar el envío de los datos al PC, pero si el usuario elige *Menú Principal* regresara al menú de inicio del dispositivo.

Figura 60. Mensaje de confirmación para la lectura de los datos a través del PC



Si el usuario eligió la opción *Iniciar Down* en el LCD del dispositivo se presentará el mensaje de la figura 61 el cual sencillamente invita al usuario a pulsar el botón Iniciar lectura de la interfaz de grafica usuario de LabVIEW que a su vez se muestra en la figura 62.

Figura 61. Mensaje de invitación a iniciar lectura desde el PC



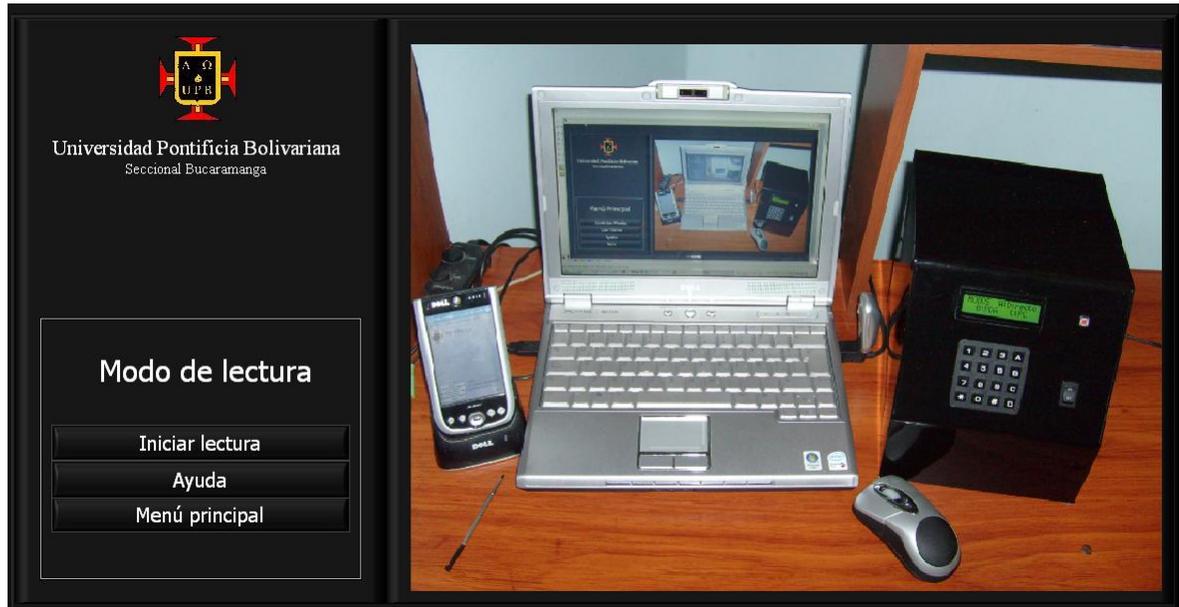
Después de configurar el dispositivo para leer los datos desde el PC, el usuario debe iniciar la lectura de los datos a través de la interfaz de LabVIEW que se muestra en la figura 62.

El usuario debe seguir los pasos que se enuncian a continuación para que la lectura de los datos se produzca con éxito.

- El usuario debe pulsar el botón *Leer Datos* de la interfaz del menú principal de la aplicación de LabVIEW.
- En la interfaz del modo lectura de datos del PC el usuario debe presionar el botón *Iniciar lectura*, para iniciar la descarga de los datos.
- Después de terminada la lectura por el PC se desplegará una ventana de dialogo en la cual el usuario debe ingresar la ruta, el nombre y la extensión con la que quiere que se guarden los datos en un archivo, se recomienda que la extensión sea *.XLS* para que los datos se guarden en una tabla de Excel y de esta forma queden mejor relacionados y mas ordenados para su posterior análisis, como se muestra en la figura 64.

- Para salir, después de terminada la lectura de los datos se debe presionar el botón *Menú Principal*.

Figura 62. Interfaz de control del modo lectura de datos con el PC



En la figura 63 se muestra el diagrama de transición de estados de la aplicación creada en LabVIEW para la lectura de los datos almacenados en el dispositivo de supervisión usando una interfaz grafica de usuario en el PC.

Figura 63. Diagrama de transición de estados del modo lectura con PC

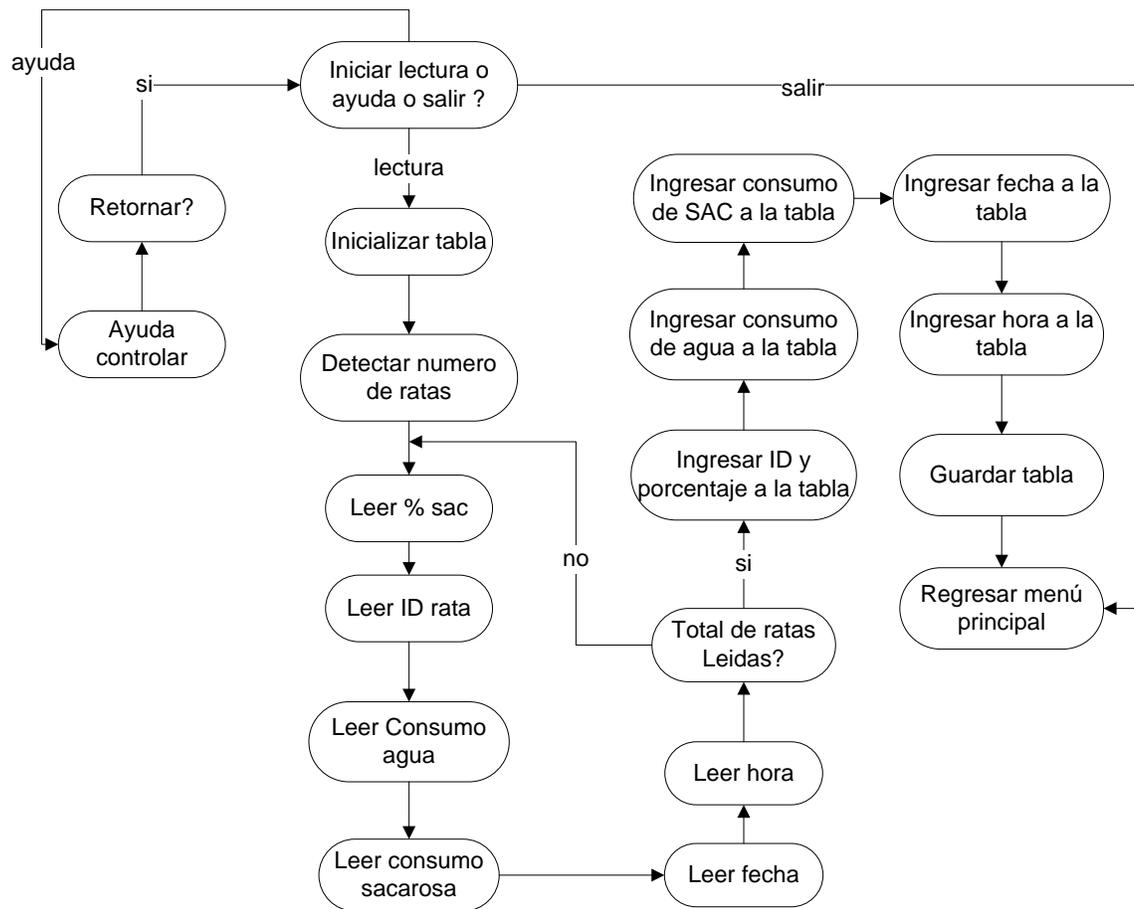


Figura 64. Tabla de resultado de la lectura de los datos almacenado en el dispositivo

PRUEBA DE CONSUMO DE SACAROSA EN RATAS

Fecha de descarga de los datos: jueves, 22 de mayo de 2008, 17:48:05

En la siguiente tabla se muestran los resultados de la prueba de consumo de sacarosa en ratas, aplicada en las fechas y horas relacionadas en la tabla .

ID de la rata	% de sacarosa	Consumo de sacarosa	Consumo de agua	Fecha DD/MM/AAAA	Hora Hor:min
ABCD	4	1	0	21/05/2000	15:00
ABCD	4	0	0	21/05/2000	15:00
AAAA	4	0	0	21/05/2025	21:54
AAAA	4	0	0	21/05/2017	21:54
AAAA	4	0	0	21/05/2025	21:54
AAAA	4	0	0	21/05/2025	21:54
AAAA	4	0	0	21/05/2025	21:54
AAAA	4	0	0	21/05/2025	21:54
AAAA	4	0	0	21/05/2025	21:54
AAAA	4	0	0	21/05/2008	21:54
AAAA	4	0	0	21/05/2008	21:54
AAAA	4	13	36	21/05/2008	21:54

Universidad Pontificia Bolivariana
Seccional Bucaramanga

4.3 MODO PDA

En el modo PDA el usuario puede controlar el dispositivo de supervisión desde un PocketPC para supervisar una o varias pruebas, con un máximo de 8 ratas por prueba. En este modo la comunicación entre el dispositivo de supervisión y la PocketPC se realiza mediante el uso de un modem Bluetooth – BlueSMiRF RP-SMA.

4.3.1 Supervisión en modo PDA

Para acceder a este modo desde el dispositivo el usuario debe elegir la opción PDA del menú principal. Después de ingresar al modo PDA se mostrará el mensaje de la figura 65, en este menú si el usuario elije la opción *Iniciar Adqui* está confirmando la supervisión de la prueba usando la PocketPC, pero si elije *menú principal* regresa el menú inicial del dispositivo.

Figura 65. Mensaje de confirmación de supervisión de la prueba en modo PDA



Después de que el usuario confirma la supervisión de la prueba usando la PocketPC, aparece el mensaje de la figura 66 en el que se invita al usuario a iniciar la supervisión de la prueba con la PocketPC presionando el botón *Iniciar prueba* de la aplicación desarrollada en LabVIEW para la PocketPC.

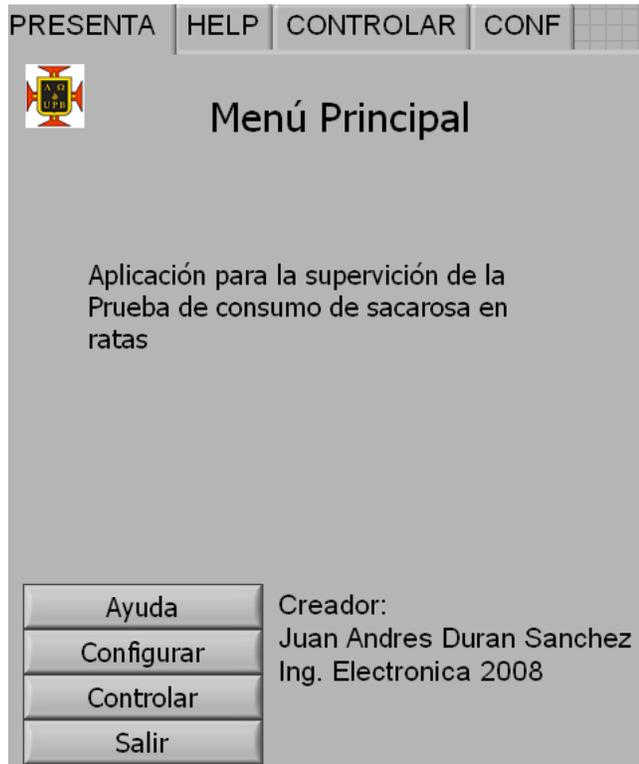
Figura 66. Mensaje de invitación a iniciar la prueba desde la PocketPC



En la figura 67 se muestra la interfaz gráfica de usuario de la aplicación desarrollada en LabVIEW para la PocketPC, en la cual el usuario a través de varios botones tiene acceso a varios submenús. Estos botones están relacionados así:

- **Ayuda:** al pulsar este botón el usuario accede a la ayuda con la que cuenta la aplicación creada en LabVIEW.
- **Configurar:** este botón remite al usuario a menú de detección y autoconfiguración de la comunicación Bluetooth entre la PocketPC y el dispositivo de supervisión.
- **Controlar:** accede al modo de control del dispositivo a través del PocketPC.
- **Salir:** como su nombre lo indica cierra la aplicación.

Figura 67. Interfaz gráfica de usuario del modo PDA



Después de configurar el dispositivo para supervisar la prueba desde la PocketPC, el usuario debe iniciar la prueba a través de la interfaz de LabVIEW que se muestra en la figura 68.

El usuario debe seguir los pasos que se enuncian a continuación para que la supervisión de la prueba se pueda realizar con éxito.

- El usuario debe pulsar el botón *Configurar* de la interfaz del menú principal de la aplicación de LabVIEW.
- En la interfaz de detección del puerto Bluetooth el usuario debe presionar el botón *Explorar y detectar*, cuando se establezca la comunicación la aplicación regresara al menú principal. (ver figura 69)
- Luego el usuario debe presionar el botón *Controlar* del menú principal para ingresar a la interfaz de la figura 68.
- En la interfaz de control el usuario debe ingresar los parámetros para realizar la prueba (duración, porcentaje de concentración de sacarosa) y

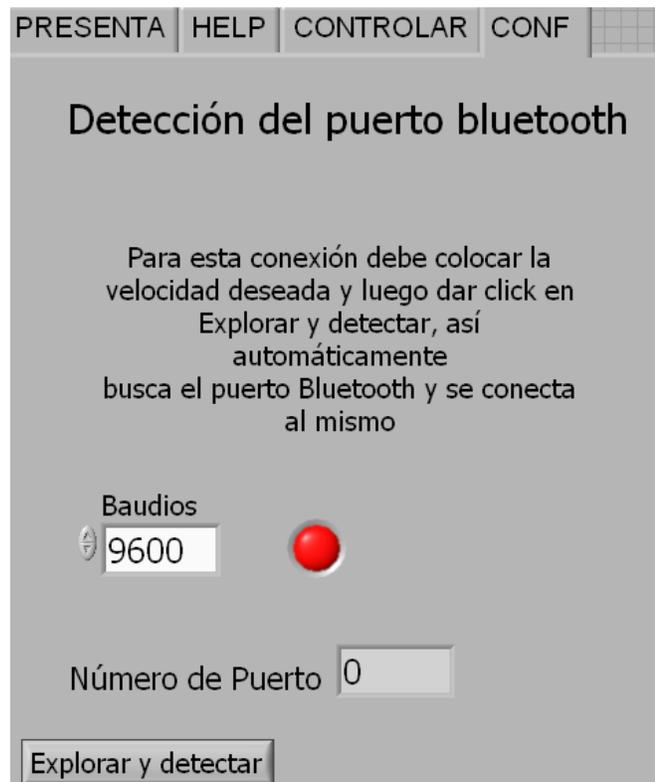
elegir las ratas a las que se supervisara y escribir su ID, después de esto el usuario debe presionar el botón *ingresar* para confirmar los datos.

- Después el usuario debe presionar el botón Iniciar para que la prueba inicie.
- Para salir, después de terminada la lectura de los datos se debe presionar el botón *Menú*.

Figura 68. Interfaz de control del modo PDA

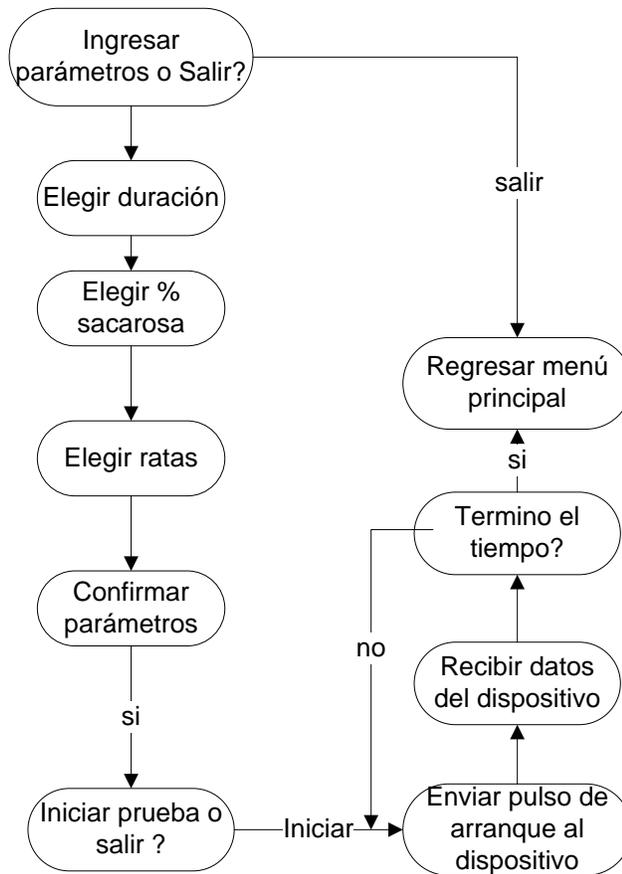
PRESENTA	HELP	CONTROLAR	CONF	
% sac		Duración 1 hora		
Id rata 1		Id rata 5		
sacarosa	0,00	sacarosa	0,00	
agua	0,00	agua	0,00	
Id rata 2		Id rata 6		
sacarosa	0,00	sacarosa	0,00	
agua	0,00	agua	0,00	
Id rata 3		Id rata 7		
sacarosa	0,00	sacarosa	0,00	
agua	0,00	agua	0,00	
Id rata 4		Id rata 8		
sacarosa	0,00	sacarosa	0,00	
agua	0,00	agua	0,00	
Ingresar	Iniciar	Menú		

Figura 69. Interfaz de de detección del puerto Bluetooth



En la figura 70 se muestra el diagrama de transición de estados de la aplicación creada en LabVIEW para la supervisión de la prueba de consumo de sacarosa en ratas usando una interfaz gráfica de usuario en una PocketPC.

Figura 70. Diagrama de transición de estados del modo supervisar con la PocketPC



5. CONCLUSIONES

Esta investigación permitió el diseño y la construcción de un dispositivo para la supervisión de la prueba de consumo de sacarosa en ratas, dicho dispositivo puede trabajar en modo directo o controlado por un PC o una PocketPC a través de una interfaz gráfica de usuario creada en LabVIEW que se comunicó con el dispositivo mediante la especificación Bluetooth.

El dispositivo de supervisión es controlado por un PIC 16F877a de la empresa Microchip, dado que este micro controlador cuenta con los módulos A/D, USART, I2C, interfaz serial asíncrona y un número apropiado de pines que permiten el desarrollo de las diferentes funciones del dispositivo.

Para el almacenamiento de los datos en el dispositivo de supervisión cuando este se trabaja en modo directo se utilizó una memoria externa EEPROM 24LC512 de Microchip. El uso de una memoria externa se hizo necesario, dado que la cantidad de datos que se requieren almacenar por prueba supera fácilmente la capacidad de almacenamiento de la memoria interna del PIC 16F877a.

El dispositivo de supervisión cuenta con una interfaz de manejo basada en un LCD 2x16 como elemento de visualización y un teclado matricial 4x4 como elemento de selección de las opciones de configuración del dispositivo. El elemento de control de la interfaz de manejo del dispositivo es el PIC 16F877a.

Se manejaron tres protocolos de comunicación en el dispositivo, uno de ellos el I2C para comunicar la memoria EEPROM 24LC512 con el PIC 16F877a, el otro el USART para comunicar el modem Bluetooth – BlueSMiRF RP- SMA y el PIC 16F877a, una interfaz serial simple para comunicar el RTC DS1304 con el PIC 16F877a. Para la comunicación del dispositivo con el PC o la PocketPC se utilizó la especificación Bluetooth mediante el modem Bluetooth – BlueSMiRF RP- SMA.

En LabVIEW 7.1 se desarrolló una aplicación para el PC para realizar la supervisión de la prueba de consumo de sacarosa en ratas y el almacenamiento de los datos propios de la prueba en un archivo plano. Esta aplicación además permite al usuario la lectura de los datos almacenados en la memoria EEPROM 24LC512 del dispositivo para ser guardados en un archivo plano en el PC. La aplicación de LabVIEW se comunica con el dispositivo mediante la especificación Bluetooth.

Para la PocketPC se desarrolló una aplicación en LabVIEW 7.1 que permite la supervisión de la prueba de consumo de sacarosa en ratas mediante una interfaz de usuario amigable. Esta aplicación se comunica con el dispositivo de supervisión mediante la especificación Bluetooth.

6. BIBLIOGRAFÍA

HERNÁNDEZ C. Carlos Gerardo – Claros L., Reynaldo - MAESTRE C., Laura Clarena. Microcontroladores de Microchip. Bucaramanga: Universidad Pontificia Bolivariana, 2004. p. 192-209, 242-246

CONDE C., Bothelo S., Berena A., Velásquez M. La vía visual puede ser el disipador de ansiogenicidad en el modelo del laberinto en cruz elevado. Bucaramanga: universidad Industrial de Santander, 2001. p. 142 – 146.

TOCCI, Ronald J. sistemas digitales principios y aplicaciones. México: Prentice Hall, 2003. p. 614 – 620