



PONTIFICIA **UNIVERSIDAD CATÓLICA** DEL PERÚ

Esta obra ha sido publicada bajo la licencia Creative Commons
Reconocimiento-No comercial-Compartir bajo la misma licencia 2.5 Perú.

Para ver una copia de dicha licencia, visite
<http://creativecommons.org/licenses/by-nc-sa/2.5/pe/>



PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



**DISEÑO DE UN SISTEMA DE MONITOREO A DISTANCIA BASADO EN
TECNOLOGÍA WEB PARA EL PROCESO DE TOSTADO DE GRANOS DE
KIWICHA**

Tesis para optar por el Título de Ingeniero Electrónico

Presentado por:

JAVIER LAZARTE PAREDES

Lima – Perú

2006

ÍNDICE

INTRODUCCIÓN.....	8
1. PROBLEMÁTICA DEL PROCESO DE TOSTADO DE GRANOS DE KIWICHA	11
1.1 INTRODUCCIÓN A LA PROBLEMÁTICA.....	11
1.2 DESCRIPCIÓN DEL PROCESO.....	12
1.3 PROCESOS PRINCIPALES QUE LO CONSTITUYEN.....	13
1.4 CONSUMIDORES FINALES.....	14
1.5 MERCADO INTERNACIONAL	15
2. DESCRIPCIÓN TEÓRICA.....	16
2.1 PRESENTACIÓN DEL PROCESO	16
2.2 TEORÍA GENERAL INVOLUCRADA EN EL PROCESO.....	17
2.3 VENTAJAS Y DESVENTAJAS.....	19
2.4 CONCEPTUALIZACIONES GENERALES	21
2.4.1 Kiwicha.....	21
2.4.1.1 Definición	21
2.4.1.2 Fisiología.....	21
2.4.1.3 Origen.....	21
2.4.1.4 Distribución	21
2.4.1.5 Ecología y Adaptación.....	22
2.4.2 Sensor	22
2.4.2.1 Definición	22
2.4.2.2 Tipos.....	22
2.4.2.3 Topología.....	23
2.4.3 WEB.....	23

2.4.3.1	Definición	23
2.4.3.2	Estándares	24
2.5	MODELO TEÓRICO.....	24
2.6	OPCIONES DE SOLUCIÓN AL ASUNTO DE ESTUDIO	26
2.6.1	Selección del Hardware	26
2.6.1.1	Sensores.....	26
2.6.1.2	Acondicionamiento de la Señal.....	28
2.6.2	Selección del Software	29
2.7	CONCLUSIONES.....	30
3.	DESARROLLO DE LA SOLUCIÓN AL PROCESO DE TOSTADO DE GRANOS DE KIWICHA.....	32
3.1	HIPÓTESIS	32
3.1.1	Hipótesis Principal.....	32
3.1.2	Hipótesis Secundaria.....	32
3.2	OBJETIVO	33
3.3	DESCRIPCIÓN DE LA SOLUCIÓN	33
3.3.1	Esquema de Distribución	33
3.3.2	Especificaciones de Hardware.....	39
3.3.3	Especificaciones de Software	41
4.	DISEÑO, PRUEBAS Y RESULTADOS	42
4.1	DIAGRAMAS DE FLUJO Y VENTANAS.....	42
4.1.1	Programa Servidor	42
4.1.2	Programa del Microcontrolador	51
4.1.3	Programa de Aplicación WEB	54
4.2	PRUEBAS Y RESULTADOS	55

4.2.1	Programa Servidor	55
4.2.2	Programa Microcontrolador	63
4.2.3	Programa de Aplicación WEB	65
4.3	DISEÑO DE LA TARJETA DE ACONDICIONAMIENTO	67
4.4	COSTOS DEL DISEÑO.....	68
CONCLUSIONES		70
RECOMENDACIONES		71
BIBLIOGRAFÍA.....		72



INTRODUCCIÓN

Desde la época prehispánica se han utilizado diversos instrumentos para la molienda o tostado de grano para su consumo, siendo el más común en México el comal, un instrumento en un inicio de barro, actualmente de metal, que se utilizaba en el proceso de tostado de grano, el cual debía estar seco. Este proceso se hace de manera manual sometiendo al fuego el comal a una temperatura de 155 °C, resultando en el tostado del grano.

En México debido al gran consumo de harina, se incursionó en la producción del grano de kiwicha, alimento natal de Sudamérica donde fue domesticada y cultivada en la época de los incas en el Perú y los aztecas en México, el cual se ha aprovechado desde tiempos prehispánicos debido a su valor medicinal, para combatir desde diarreas mediante la infusión de las hojas de la plata hasta gelatina con alto contenido de fibra, elaborada en base a kiwicha, para combatir el cáncer de colon, así como otras enfermedades como osteoporosis, diabetes, obesidad, etc.

Por otro lado la producción del grano de kiwicha da lugar a diversos productos finales como son alegrías, kiwicha (cereal) tostado, granolas, tamales, atoles, pinole, mazapán, etc., tanto así que se formó *Amarantum* una asociación Mexicana del Amaranto, la cual elabora los productos antes mencionados. Esto lo hace por medio de un proceso simple, para la elaboración de la harina, el cual consiste en el paso del grano por un molino el cual separa el grano molido, en dos partes de acuerdo al producto final que se requiere, ya que más adelante se le incorporarán otros productos. Por otro lado para el tostado de grano lo hace por una máquina en la cual

ingresa el grano por un costado y es tostado a presión. Estos productos también son elaborados en diversas partes del mundo como son Estados Unidos, Checoslovaquia, Kenia, China y en América Latina.

En la actualidad la mayor parte de la industria nacional la conforman las pequeñas y medianas empresas (PYME) que elaboran una diversidad de productos, los cuales en muchos casos son procesos, en la medida de lo posible, manuales. Uno de los rubros en los que incursionan estas empresas es la producción de alimentos, en un caso más particular, la de los alimentos elaborados en base a granos de kiwicha, los cuales tienen una gran demanda a nivel nacional e internacional. Como es sabido, en nuestro país, en tiempos muy antiguos el consumo de cereales y granos era considerado ingrediente fundamental en toda comida debido a su gran valor nutritivo y buena conservación en diversos climas. Esta tradición dietética se sigue conservando hoy en día y la encontramos en diversas formas, desde el grano natural en desayunos hasta un producto más elaborado como son en dulces embasados.

Existen dos formas principales de elaboración del grano tostado de kiwicha, el primero es el de reventar el grano sometiéndolo a presión en una cámara preparada adecuadamente, el segundo, y el más usado, es el de reventar el grano tostándolo, este proceso se puede hacer de forma manual en una olla de barro o metal sometiéndola al fuego y revolviendo de manera constante, o de manera mas industrial, haciendo pasar el grano a través de un tubo de metal giratorio sometido al fuego. En este último caso se requiere de un continuo monitoreo de temperatura, ya que una elevación de esta daría

como resultado una mala producción y por ende una caída de la misma. Este y otros factores como por ejemplo el nivel del producto, son necesarios monitorear y controlar para tener una buena producción.

El problema principal que presentan las pequeñas y medianas empresas es el control del proceso de producción, el cual en la mayoría de los casos se da de una manera totalmente manual, ocasionando, por falla del hombre o mal control de la maquinaria usada, una baja en el proceso de producción. Esto se da, por una parte, a que no se cuentan con los recursos necesarios para implementar un sistema de monitoreo y/o control de las variables críticas de la producción, ya que esto implica requerir de los servicios de un ingeniero o técnico experimentado, y dicho proceso al poder controlarse de alguna manera rudimentaria, no se toma en cuenta.

Todo este monitoreo del proceso resulta en el almacenamiento de datos, los cuales deben ser visualizados y distribuidos en las áreas pertinentes para un mejor control.

El objetivo principal del objeto de estudio es el de mejorar la calidad de producción de granos, mediante un adecuado monitoreo y control de las variables del proceso y una distribución de los datos del mismo.

1. PROBLEMÁTICA DEL PROCESO DE TOSTADO DE GRANOS DE KIWICHA

1.1 Introducción a la Problemática

Si bien el proceso de tostado del grano de kiwicha se hace con el mismo principio que se hacía en tiempo anteriores, calentando el grano y revolviendo, ahora se realiza con maquinaria más sofisticada, sin embargo se requiere en gran porcentaje de la mano del hombre, de las decisiones que tome y esto conlleva a errores y por ende a una baja de la producción.

En nuestro país, donde la mayoría de las empresas están conformadas por las que entran en el sector de pequeñas y medianas empresas (PYME), este tipo de proceso se realiza de manera rudimentaria preocupándose principalmente por el proceso del tostado del grano sin contar con un debido monitoreo de las variables críticas del proceso, y un sistema básico de alarma, lo que ayudaría de manera mas precisa la supervisión del proceso por parte de los operadores.

Centrándome en el proceso de tostado del grano de kiwicha, se tiene como variable crítica a la temperatura, la cual si bien se mantiene constante durante todo el proceso, no se tiene, en el caso antes mencionado de las PYME, un monitoreo de dicha variable teniendo como medición en muchos casos de la temperatura a un termómetro de mercurio de escasa precisión o al producto final, ya que según su consistencia se ve el grado de temperatura que se debe variar. En este proceso también se tiene como variable al nivel, el cual debe medir e indicar la cantidad de producto tanto a la entrada como a la salida del proceso, ya que una escasez de producto a

la entrada del proceso deja a la máquina trabajando sin generar una producción y en cuanto a la salida del proceso el producto se recibe en recipientes, los cuales no deben exceder su capacidad ya que se tendría una pérdida de producto.

Todos estos parámetros solo son medidos en el lugar donde se realiza el proceso de producción, no teniéndose un monitoreo mas amplio refiriéndome a una cuestión geográfica, es decir que estos parámetro puedan ser visualizados fuera del lugar de trabajo.

Un adecuado monitoreo de los parámetros antes mencionados, ayuda a una adecuada supervisión del proceso y esto a una mejora del producto final y por ende ayuda a que si se quiere entrar en un mercado tan competitivo como el nuestro, debido a que por cada producto se tienen muchas marcas diferentes, se tiene que sobresalir tanto en precio como en cuestión a la calidad del producto.

1.2 Descripción del Proceso

El tostado de granos de kiwicha, se constituye principalmente por tres procesos principales, los cuales son: la selección inicial del grano, el proceso de tostado propiamente dicho y el proceso de selección del grano tostado.

De estos tres, el proceso de tostado es el objeto del asunto de estudio, ya que en cuanto a la metodología el primer y tercer proceso, el de selección del grano crudo y tostado, se hace en cierta forma de manera manual sobre todo el primero ya que el grano que llega se vacía en un cernidor separando el grano bueno, esto tomando unicamente en cuenta el tamaño del grano.

En cuanto al personal encargado de los procesos, se cuenta con un personal técnico, más que todo personal basado en la experiencia previa del proceso, ya que como indicamos anteriormente, este es un proceso que se hace también y comúnmente de manera rudimentaria, completamente manual. El espacio en el cual se trabaja, es un lugar de trabajo amplio pero sin separación de áreas.

1.3 Procesos Principales que lo Constituyen

En cuanto a la descripción del proceso de tostado en sí, es la siguiente:

Primero se verifica si hay producto en la tolva de ingreso (grano crudo), si no lo hubiera se llena la tolva hasta su altura máxima. Luego de verificado que haya producto en la tolva de ingreso, se procede a verificar si los recipientes de salida, donde se almacena el grano ya tostado, están por debajo del nivel medio, dependiendo de cuanta cantidad de producto se vaya a tostar ese día.

Luego se prende el quemador del horno, el cual debe llegar a una temperatura exacta para comenzar el proceso. A continuación se abre la línea de ingreso de granos, y en un eventual quemado de grano en el horno, se procede a desprender dicho grano de las paredes del horno.

De todos estos procesos se pueden describir cuales son los hechos que acarrear problemas, a continuación se detallan éstos:

Se requiere personal presente todo el día.

Para el proceso de producción se debe contar siempre con producto en la tolva de ingreso, ya que sino la máquina trabaja sin producto, y además se debe verificar que los recipientes de almacenamiento no estén totalmente llenos ya que el producto saliente se puede derramar. Por este motivo una persona debe controlar esto todo el tiempo.

Tiempo de espera aproximado basado en experiencia.

Al encender el horno donde se va a reventar el grano, se debe esperar un tiempo para que éste caliente y el grano ingrese con una temperatura adecuada del horno (precalentado). Este tiempo es estimado, por lo que se espera un tiempo prudencial.

Proceso se realiza manualmente.

Para controlar el quemador del horno se hace uso de una válvula. Para el control de dicha llama se hace el cierre o apertura de la válvula manualmente.

1.4 Consumidores Finales

En cuanto al abastecimiento del grano, éste se da por medio de un proveedor, el cual trae granos de diferentes provincias del Perú principalmente de Huaraz, Arequipa y Cusco.

Para la venta a los diferentes centros finales de consumo, como son las casas naturistas, que constituyen el principal porcentaje de ventas, las

bodegas y los consumidores individuales, se tiene como medio regulador al ministerio de salud el cual solicita para una adecuada venta de alimentos que se cuente con el certificado de salud correspondiente. En cuanto a los competidores locales tenemos otros centros de producción de granos de kiwicha, que en nuestro país son principalmente pequeñas y medianas empresas.

1.5 Mercado Internacional

En cuanto al mercado internacional, éste si bien constituye una gran demanda, también requiere gran volumen de exportación, no dándose abasto la producción de una pequeña o mediana empresa.

Además a esto se suma la cantidad de impuestos y regulaciones que se debe pagar y tener en cuenta para ingresar a un mercado internacional.

2. DESCRIPCIÓN TEÓRICA

2.1 Presentación del Proceso

El monitoreo de diversos parámetros físicos es una tecnología que se ha ido desarrollando a través de los años, con tal diversidad que ahora podemos encontrar una gran cantidad de sensores que registran una gran variedad de parámetros necesarios para un control eficiente de un proceso o procesos determinados. Sin embargo dicho monitoreo de variables físicas deben ser registradas para un posterior control o una simple visualización de las mismas.

Hoy en día el compartir información o simples datos entre diferentes lugares se ha vuelto una necesidad básica, por lo tanto esta tecnología se ha incorporado con el propósito de compartir los datos registrados a diferentes usuarios permitidos, sin que éstos tengan que ir personalmente a leer o tomar datos sino que estos datos le son proporcionados en su computador.

Actualmente este sistema esta siendo utilizado en una gran cantidad de industrias que cuentan con una amplia línea de producción (alimentos, bebidas, etc.) o empresas que cuentan con una gran variedad de productos (inventario de los diversos tipos de productos que venden).

Este sistema principalmente es tratado mediante un software de aplicación, el cual será el nexo entre el usuario y el valor del parámetro físico medido. Así el usuario podrá ver todo el proceso de manera segura y práctica.

En el presente documento se presentarán los actuales avances en el desarrollo de tres partes básicas, el monitoreo de variables físicas, el

almacenamiento de datos y los tipos de comunicación de los datos almacenados a diferentes terminales.

2.2 Teoría General Involucrada en el Proceso

SENSORES

“Las aplicaciones de la electrónica, presentes actualmente en innumerables aspectos de nuestra vida cotidiana, no serían posibles sin los sensores. Sin la capacidad que éstos ofrecen de medir magnitudes físicas para su conocimiento o control, muchos de los dispositivos electrónicos no serían más que simples curiosidades de laboratorio.” [4]

“Las diversas formas de automatizar los procesos y servicios se realizan a través del uso de sensores, controladores y actuadores facilitando la producción y minimizando los recursos humanos.” [5]

“La reducción del precio de los procesadores (digitales) de información ha extendido las aplicaciones de los microprocesadores (μP) y de microcontroladores (μC), y esto ha conllevado un auge de la telemetría de señales digitales. En algunos casos los sensores han pasado de ser un componente mecánico aislado a ser un subsistema electrónico capaz de comunicarse con un sistema basado en microprocesadores, o integrado en dichos sistemas y con capacidad de autodiagnóstico.” [4]

“El avance en el conocimiento del silicio y otros semiconductores, y en las tecnologías de la fabricación de microcircuitos, ha permitido integrar cada vez más funciones en una misma oblea o en microcircuitos con soporte

común. Esto ha llevado a sensores que realizan funciones adicionales a la mera obtención de una señal a partir de una magnitud física, y así facilitan el control distribuido. Estos sensores se denominan, genéricamente, sensores inteligentes.” [4]

TOPOLOGÍA

La topología de la red es el método para interconectar los equipos y sistemas conectados a ella así como la forma que adoptan. La topología depende del sistema de control que se utilice y el cableado en función de los requerimientos del sistema.

ESTRELLA

La red de estrella es la conexión utilizada por los sistemas centralizados donde existe un único controlador sobre el que pasa toda la información. [6]

Algunas características de esta topología son:

- Permite una comunicación rápida.
- Exige un nodo central de gran capacidad.
- Estructura simple.
- Especialmente apta cuando los terminales se encuentran en un área geográfica concentrada y el número no es demasiado grande. Si el área es muy grande puede que merezca más la pena un anillo, porque se ahorra cable.
- Fiabilidad baja, mucha dependencia del nodo central.

- La reconfiguración de la red puede resultar problemática si es necesario detener el nodo central ya que en ese caso se interrumpe el servicio durante un tiempo. [7]

MALLA

Redes diseñadas sin un nodo o equipo terminal central en la que todos están conectados entre sí, de manera que ninguno posee una función especial en cuanto a la concentración de tráfico. Se muestra en la figura 7. Puede utilizarse para comunicar solamente equipos terminales, así se consigue tener todos los terminales interconectados de forma permanente, o bien conectar equipos conmutadores para dar robustez a la red, ya que se cuenta con caminos alternativos en caso de fallo en algún nodo. Al estar cada estación unida a las demás existe independencia entre ellas y redundancia de caminos que confiere mayor fiabilidad a la red. También puede existir el mallado parcial. [7]

BUS

Redes en las cuales cada equipo terminal está conectado a un bus (medio de transmisión compartido) por el que circula toda la información que tiene como destino un equipo de esa red. [7]

2.3 Ventajas y Desventajas

“Para sacar provecho del bajo coste de los μ C, lo mejor es a veces emplear una interfaz directa sensor- μ C, es decir, sin convertidor A/D (CAD).” [4]

“La repercusión inmediata de los sensores inteligentes en un sistema de medida y control es que reducen la carga sobre controladores lógicos programables (PLC), PC u otros controladores digitales, aparte de aumentar la fiabilidad del sensor”. [4]

Son numerosas las ventajas que aporta la conexión en red local; destacamos como las más importantes las siguientes:

- Mantener bases de datos actualizadas instantáneamente y accesibles desde distintos puntos.
- Facilitar la transferencia de archivos entre miembros de un grupo de trabajo.
- Compartir periféricos caros (impresoras láser, plotters, discos ópticos, etc.)
- Disminuir el costo de software comprando licencias de uso múltiple en vez de muchas individuales.
- Mantener versiones actualizadas y coherentes del software.
- Facilitar la copia de respaldo de los datos.
- Correo electrónico.
- Comunicarse con otras redes (bridges y routers).
- Conectarse con minis y mainframes (gateway).
- Mantener usuarios remotos vía módem. [3]

“Actualmente ya coexisten sistemas con comunicación simultánea analógica (4-20 mA) y digital, pero la mayoría de los sistemas actuales se basan en el bucle de 4-20 mA. Por razones de coste, ninguno de estos sistemas desaparecerá.” [4]

2.4 Conceptualizaciones Generales

2.4.1 Kiwicha

2.4.1.1 Definición

La kiwicha, como se le conoce en el Perú, pertenece a la familia de las AMARANTHACEAE y su nombre científico es *Amaranthus caudatus*. [8]

Nombre común:

Amaranto, kiwicha, coyo, achis, achita, qamaya, sangoracha, coyo, achis, qamaya, coimi, millmi, jataco.

2.4.1.2 Fisiología

Planta anual de 1 a 2 metros y medios de alto, con inflorescencias grandes y generalmente muy coloridas, de hojas pecioladas, ovales, opuestas o alternos, de color verde o púrpura. [9]

2.4.1.3 Origen

La kiwicha es un grano originario de América del Sur, donde fue domesticado. Fue comúnmente cultivada durante el tiempo de los Incas en el Perú y de los aztecas en México. La kiwicha se cultiva también en Bolivia, Ecuador y Argentina. [10]

2.4.1.4 Distribución

Se cultiva desde el sur de Ecuador, a través de Perú y Bolivia hasta el noroeste de Argentina

2.4.1.5 Ecología y Adaptación

Su adaptación a zonas agroecológicas muy diferentes permite que se cultiven desde el nivel del mar hasta los 3400 msnm, se extiende desde Ecuador hasta el norte de Argentina, en áreas templadas y valles interandinos.

En el Perú:

En el Perú se cultiva en la zona agroecológica Quechua, valles interandinos de Cajamarca, Ancash, Ayacucho, Huancavelica, en el valle de Urubamba y en el valle de Majes de Arequipa (en alturas entre 1500 y 3500 msnm) en zonas donde se produce también el maíz.

Es sensible al frío, pudiendo soportar sólo 4 °C al estado de ramificación y 35 - 40 °C como temperatura máxima. [10]

2.4.2 Sensor

2.4.2.1 Definición

Dispositivo que detecta una determinada acción externa, temperatura, presión, etc., y la transmite adecuadamente. [8]

2.4.2.2 Tipos

Analógicos:

Los sensores analógicos tienen principalmente 2 valores de corriente con su equivalencia en voltaje:

- 0 – 20 mA 0 – 10 V
- 4 – 20 mA 1 – 5 V

Digitales:

Los sensores digitales tienen salidas de dos valores constantes 1 (5V) y 0 (0V) lógico.

2.4.2.3 Topología

Estrella:

La red de estrella es la conexión utilizada por los sistemas centralizados donde existe un único controlador sobre el que pasa toda la información

Anillo:

En la red de anillo cada controlador está conectado a otros dos, y así sucesivamente, formado un anillo.

Bus:

La red en bus es una arquitectura donde todos los elementos conectados a ella tengan la estructura de controladores, y que sean conectados al bus.

2.4.3 WEB

2.4.3.1 Definición

World Wide Web (WWW), es un sistema de navegador web, como una página, sitio o conjunto de sitios que proveen información por los medios descritos

2.4.3.2 Estándares

Localizador Uniforme de Recursos

(URL) Especifica cómo a cada página de información se asocia una "dirección" única en donde encontrarla.

Protocolo de Transferencia de Hipertexto

(HTTP) Especifica cómo el navegador y el servidor intercambian información en forma de peticiones y respuestas.

Lenguaje de Marcación de Hipertexto

(HTML) un método para codificar la información de los documentos y sus enlaces.

2.5 Modelo Teórico

Para desarrollar este sistema es necesario el uso de diferentes herramientas como son software de aplicación, necesario para la visualización de los datos enviados por los sensores. También es necesario conocer la fisiología de la kiwicha, para saber como se comporta frente al proceso de tostado del grano, conocer la temperatura a la cual revienta el grano para así mantener dicho parámetro constante durante el proceso de fabricación.

También es necesaria la distribución de la información dada por los sensores, esto mediante la tecnología WEB la cual proveerá a los equipos terminales de dicha información haciendo uso de un software específico.

Parámetros Principales

En el proceso de producción se debe tomar en cuenta diversos parámetros como son los que se describen a continuación:

- Posicionamiento de sensores. Se debe ubicar los sensores de tal manera que realicen la toma de datos de los valores reales concernientes al parámetro físico que se requiere (temperatura, nivel, etc.).
- Control del proceso. Mejorar el producto final, a fin de elevar el nivel de producción reduciendo, gracias a un control, el daño de la producción (grano tostado quemado).
- Nivel de producción. Este indicador se refiere al nivel de producción que se debe mantener para satisfacer la producción diaria según un historial de ventas.
- Tiempo de tostado del grano. Este indicador se refiere al tiempo que es necesario para reventar el grano de kiwicha, y así poder determinar la temperatura apropiada a mantener en el proceso de acuerdo a la maquinaria usada.
- Temperatura de tostado del grano. Monitoreo de la temperatura del proceso, se debe mantener a una temperatura constante para evitar que se queme el grano.
- Costos de mantenimiento. Se refiere a los costos por mantenimiento del sistema.

2.6 Opciones de Solución al Asunto de Estudio

A continuación se detallarán las posibles soluciones para la realización del asunto de estudio.

2.6.1 Selección del Hardware

En cuanto a la selección del hardware, se detalla a continuación los requerimientos técnicos que deben tener los equipos.

2.6.1.1 Sensores

Sensor de temperatura

El cual se encarga de enviar los datos de los valores de temperatura de la máquina encargada del tostado del grano.

Entre los sensores que se han tomado en cuenta están:

- **Termocuplas**

Poseen un rango nominal de temperatura entre -150 a 1500 °C, tiene un costo bajo y una linealidad alta, pero el grado de precisión no es muy bueno.

- **PT-100**

Consiste en un alambre de platino que a 0 °C tiene una resistencia de 100 ohms y que al aumentar su temperatura aumenta su resistencia eléctrica. Poseen una buena linealidad (Figura 2.1) y los valores se

pueden obtener de tablas (Ver Anexo 1). Superan a otros tipos de sensores en aplicaciones con temperaturas entre -100 a 200 °C [12].

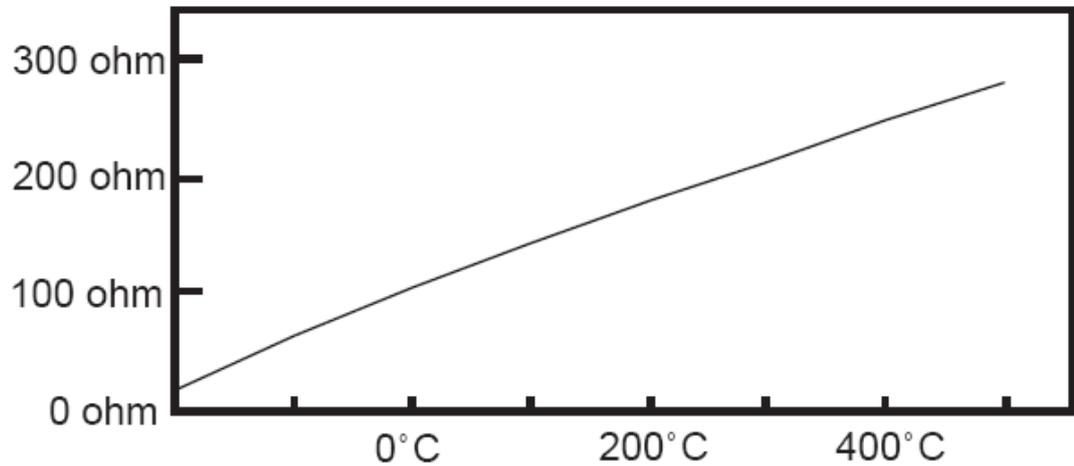


Figura 2.1. Gráfico Resistencia vs. Temperatura.

Sensor de nivel

El cual se encarga de tomar el dato del nivel de producto tanto a la entrada como a la salida de la máquina de tostado.

Entre los sensores que se tomaron en cuenta están:

- **Ultrasónico**

Estos sensores funcionan emitiendo y recibiendo ondas de sonido de alta frecuencia. La frecuencia generalmente es de aproximadamente 200 KHz, un valor demasiado alto para ser detectado por el oído humano. [11]

- **Fotoeléctrico**

Los sensores fotoeléctricos trabajan detectando el cambio en la cantidad de luz que, o bien es reflejada, o bien interrumpida por el objeto a detectar (diana). El cambio en el haz de luz puede ser el resultado de la presencia o ausencia de la diana, o el resultado de un cambio en el tamaño, perfil, receptividad o color de dicha diana. [11]

2.6.1.2 Acondicionamiento de la Señal

Para el acondicionamiento de la señal proveniente de los sensores se presentan dos posibles soluciones:

Tarjeta de adquisición de datos (DAQ)

Esta tarjeta acondiciona señales estándar de sensores (4 – 20 mA, 0 – 5 V), a valores digitales que pueden ser interpretados por un software específico de aplicación el cual se mencionará en la parte de *selección del software*.

Circuito de Acondicionamiento de Señal

Es un circuito provisto principalmente por un microcontrolador ATmega8L, el cual se encarga del manejo, muestreo y digitalización de la señal de los sensores previamente acondicionada a valores analógicos (0 – 5 V).

Estos valores digitales serán enviados al software encargado del monitoreo mediante comunicación serial (RS-232).

2.6.2 Selección del Software

Para la selección del software encargado del monitoreo de las variables del proceso se cuenta con dos posibles soluciones las cuales se describe a continuación:

LabView

Es un software de la empresa National Instruments, el cual junto con el hardware antes mencionado (tarjeta DAQ), provee un monitoreo de las señales provenientes de dicho hardware. Este software cuenta con un entorno gráfico tanto para el desarrollo del programa como para la visualización del proceso.

Java

Es un lenguaje de programación sobre el cual se implementa el software encargado del monitoreo de las señales provenientes del circuito de acondicionamiento de señal descrito como la segunda opción de hardware a usar.

Microsoft Visual Web Developer 2005 Express Edition

Es la plataforma sobre la cual se va a implementar el programa de aplicación Web, la cual soporta intercambio de información con diferentes bases de datos entre ellas SQL Server Express 2005, la cual servirá de nexo entre el programa de aplicación del servidor y el de Web.

2.7 Conclusiones

Como solución al problema planteado anteriormente, se decidió por el uso de la segunda opción tanto en cuestión de hardware como de software, es decir de la tarjeta de acondicionamiento de señal provisto del microcontrolador ATmega8L y el lenguaje de programación Java para el monitoreo de las variables.

En cuanto a la selección del hardware, se optó por esta segunda opción debido al precio del mismo ya que para cuestiones de monitoreo de señales como las provistas por los sensores industriales, el microcontrolador ATmega8L presenta una buena opción ya que cuenta con un comprador análogo-digital interno, el cual muestrea las señales con una buena resolución y velocidad, y las traduce en valores digitales para ser enviadas vía serial al software encargado del monitoreo.

En cuanto a la selección de los sensores, para el sensor de temperatura se optó por un sensor de temperatura tipo PT-100 debido a que son dispositivos con un comportamiento altamente lineal y para la temperatura de aplicación del proceso en estudio (180 °C) es recomendable y además poseen un mayor grado de resolución que las termocuplas (0.1 °C).

Con respecto a los sensores de nivel, los seleccionados fueron los sensores de nivel fotoeléctricos debido a que su costo es menor con respecto a los sensores de nivel por ultrasonido. Además proveen una detección muy exacta y es utilizado de manera industrial. Debido a la irregularidad de la superficie que se va a medir (tolva con granos), es necesario utilizar el tipo

de sensor difuso normal, el cual detecta la luz difundida en varias direcciones debido a la irregularidad de la superficie.

Ahora para la selección del software, se hizo uso del lenguaje de programación Java, el cual al ser un software libre no requiere de un pago de licencias para distribuir los programas de aplicación finales. Y además posee la capacidad de interactuar con dispositivos externos vía serial (RS-232).



3. DESARROLLO DE LA SOLUCIÓN AL PROCESO DE TOSTADO DE GRANOS DE KIWICHA

3.1 Hipótesis

3.1.1 Hipótesis Principal

Debido a que en las pequeñas y medianas empresas (PYME), en el sector de producción de alimentos en base a granos de kiwicha, cuentan con un rudimentario sistema de monitoreo de las variables del proceso de producción, afectando a ésta, en cuanto a la calidad, entonces la implementación de un sistema capaz de monitorear adecuadamente dicho proceso dando facilidad y confiabilidad a la tarea del operador, permitirá una mejor calidad del producto final, dando una mayor producción efectiva, así como también un control de la seguridad misma de la maquinaria.

3.1.2 Hipótesis Secundaria

El uso de un sistema de monitoreo del proceso de producción de granos de kiwicha, el cual involucra la decisión total del operador para dicha supervisión, hace difícilmente posible la obtención de una producción de calidad, limitando a dicha producción a un mercado reducido como bodegas o pequeñas casas naturistas.

Por eso el uso de un adecuado sistema de monitoreo tendrá como resultado la buena calidad del producto, pudiendo entrar a un mercado más competitivo y amplio, y además reducir costos ya que se contará con una producción más efectiva.

3.2 Objetivo

General

Implementar un sistema de monitoreo para la producción de granos de kiwicha y distribución de datos vía WEB.

Específico

Precisar los puntos críticos del proceso de producción de granos de kiwicha, las variables del proceso a tomar en cuenta.

Realizar un adecuado sistema de monitoreo, con el objetivo de dar precisión al proceso antes mencionado.

Realizar pruebas al sistema planteado, a fin de comprobar la efectividad del mismo.

3.3 Descripción de la Solución

3.3.1 Esquema de Distribución

El esquema general de distribución tanto del hardware como los equipos terminales es como el que se muestra en la Figura 3.1.

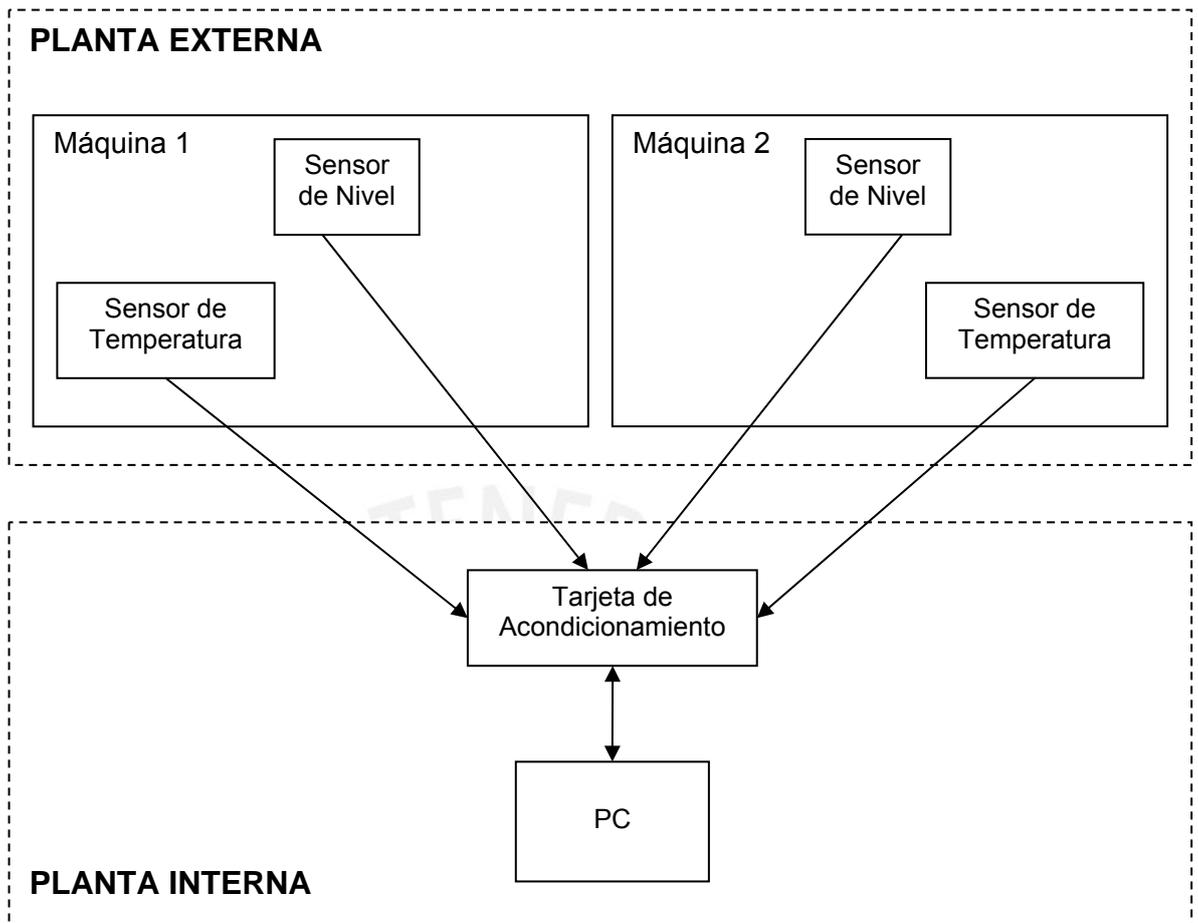


Figura 3.1. Gráfico de las señales entre los diferentes módulos.

También se muestra la distribución de los sensores (Figura 3.2) en cada máquina y en las tolvas tanto de entrada como de salida.

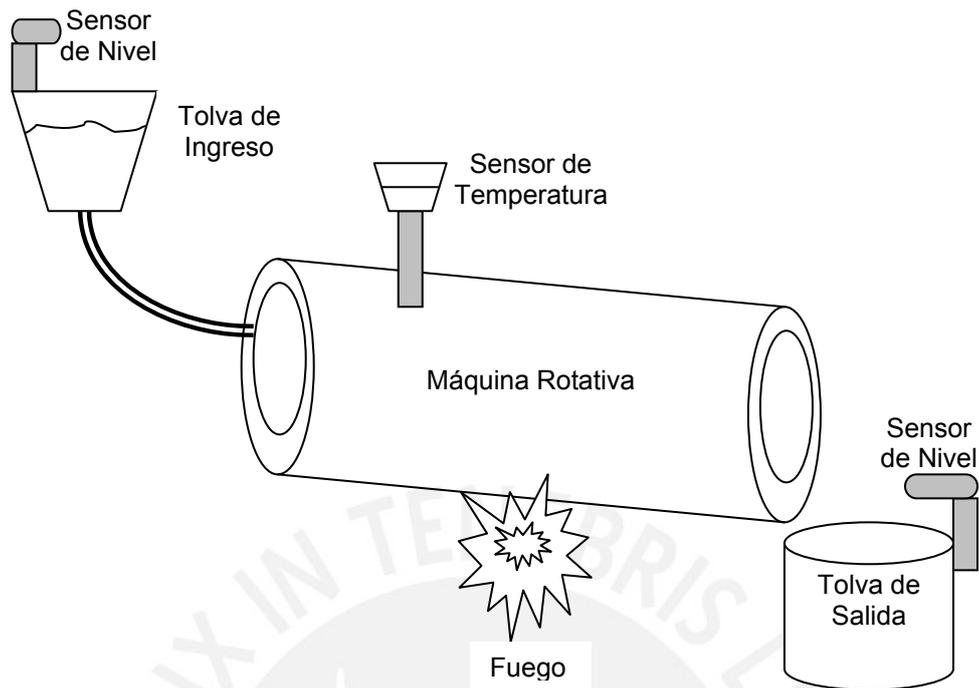


Figura 3.2. Distribución de sensores.

Además se muestra una fotografía de la tolva de entrada (Figura 3.3), la tolva de salida (Figura 3.4) y la máquina encargada del proceso de tostado (Figura 3.5).



Figura 3.3. Tolva de ingreso de producto.



Figura 3.4. Tolva de almacenamiento de producto final.



Figura 3.5. Máquina tostadora.

Teniendo en cuenta el plano de distribución de áreas del local (Figura 3.6), el plano con las medidas del área en el cual se va a realizar el proceso (Figura 3.7), se toman las medidas y condiciones del cableado.



Figura 3.6. Plano de distribución y medidas de áreas del local donde se realiza el proceso de tostado.

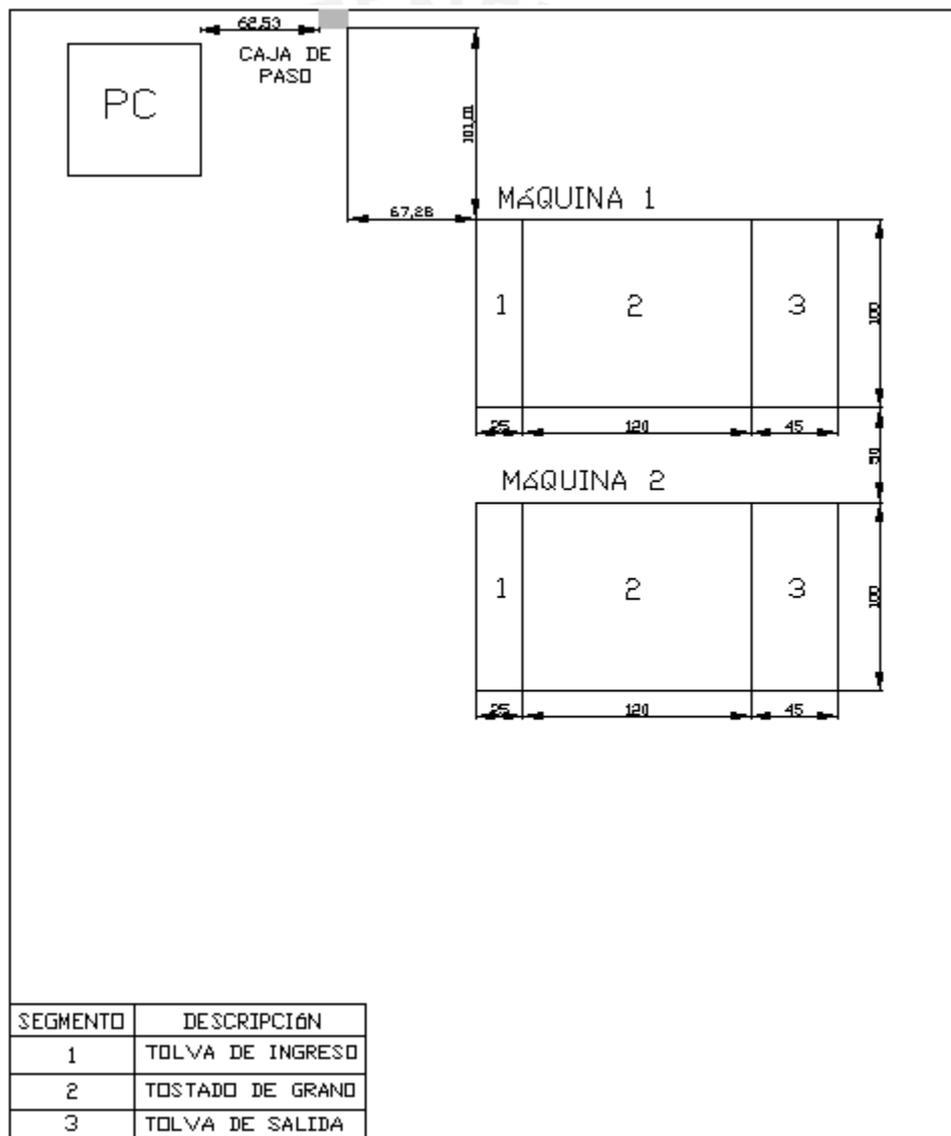


Figura 3.7. Plano de distribución y medidas de áreas de proceso de tostado.

En cuanto a las medidas de los cables, tomando las medidas descritas en la figura 3.7 y teniendo en cuenta las siguientes alturas, de la ubicación de los sensores y caja de paso al suelo:

Tolva de ingreso: 2.25 metros.

Máquina de tostado: 1.5 metros.

Tolva de salida: 1 metros.

Caja de paso: 2 metros.

Se obtienen las siguientes medidas de cable (Tabla 3.1) hasta la entrada de la caja de paso:

Sensor	Longitud (metros)
Máquina 1	
Nivel (tolva de ingreso)	6.92
Temperatura (máquina de tostado) (*)	7.02
Nivel (tolva de salida)	7.57
Máquina 2	
Nivel (tolva de ingreso)	7.42
Temperatura (máquina de tostado) (*)	7.52
Nivel (tolva de salida)	8.07

Tabla 3.1. Medidas de cables de cada sensor.

(*) Se tiene en cuenta que el sensor de temperatura estará ubicado en el punto medio de la máquina.

3.3.2 Especificaciones de Hardware

Teniendo en cuenta lo antes mencionado se procedió a la selección específica de los equipos, cuyas características técnicas principales se detallan a continuación.

Sensor de Temperatura

Para la medición de la temperatura del proceso se seleccionó un PT-100 de las siguientes características:

- Rango de medida: $-20 - 200$ °C
- Longitud del sensor: 250 mm
- Temperatura ambiente permitida: $-20 - 70$ °C

La selección de este sensor fue debido a que cuenta con una buena precisión de temperatura, aproximadamente 0.1 °C hasta una temperatura de 200 °C, más allá que de la de operación y la del valor máximo.

Sensor de Nivel

En cuanto a la medición del nivel tanto a la entrada como a la salida del proceso, tolva de entrada y tolva de salida (*Ver Anexo 2*), se seleccionaron los siguientes equipos con las características indicadas a continuación:

- Rango de medida: 20 – 80 cm (tolva de ingreso)

20 – 130 cm (tolva de salida)

- Precisión: ± 1 mm
- Temperatura ambiente permitida: -20 – 70 °C
- Voltaje de alimentación: 24 VDC
- Rango de salida: 4 – 20 mA

Circuito de Acondicionamiento

El circuito de acondicionamiento va a estar conformado principalmente por el microcontrolador ATmega8L, el cual se encargará del muestreo y digitalización de los valores analógicos proveniente de los sensores, el envío de datos de manera serial (RS-232) a la computadora con el programa principal y las salidas de control digital (*Ver Anexo 3*). En cuanto a las especificaciones de los comparadores análogo digital interno del ATmega8L se tiene:

- Numero de ADC: 6
- Resolución: 8 bits
- Voltaje de Entrada: 0 – 5 V
- Tiempo de Conversión: 13 – 260 us
- Muestreo: 15 kSPS (máximo)

3.3.3 Especificaciones de Software

El programa de aplicación encargado del monitoreo, esta desarrollado sobre el lenguaje de programación Java de Sun Microsystem, que presenta un entorno gráfico para la visualización de los parámetro específicos.

Para el programa de aplicación encargado de la parte de la distribución de datos vía WEB será desarrollado en Microsoft Visual Web Developer 2005 Express Edition.

Estos dos programas compartirán una base de datos, la cual será SQL Server Express 2005 la cual tiene una capacidad de almacenamiento de 4Gb.

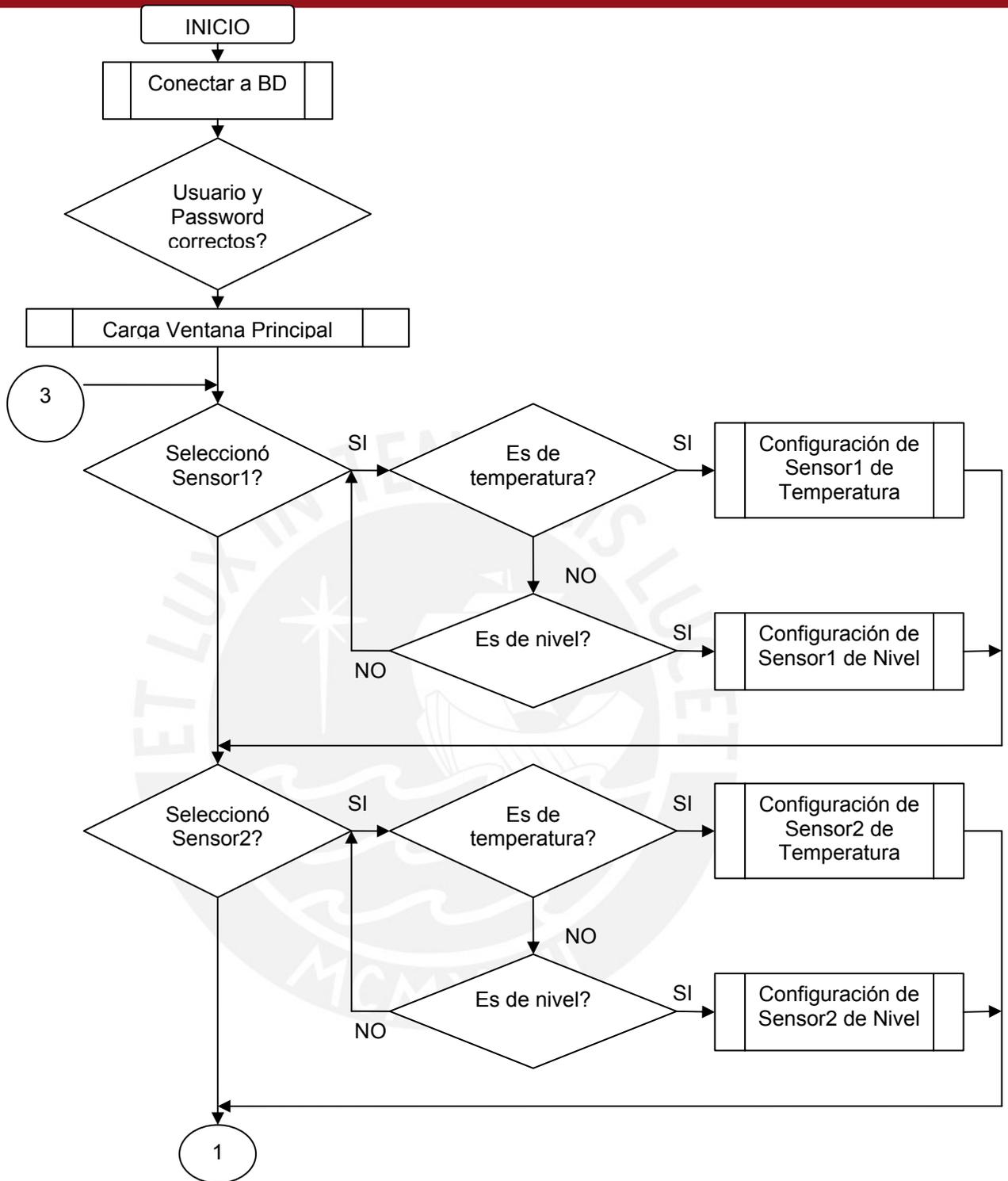
Para el funcionamiento del programa servidor se deberá alquilar un espacio en un Web Hosting donde estará almacenado el programa WEB, y además adquirir un dominio (dirección web) para acceder a dicho programa.

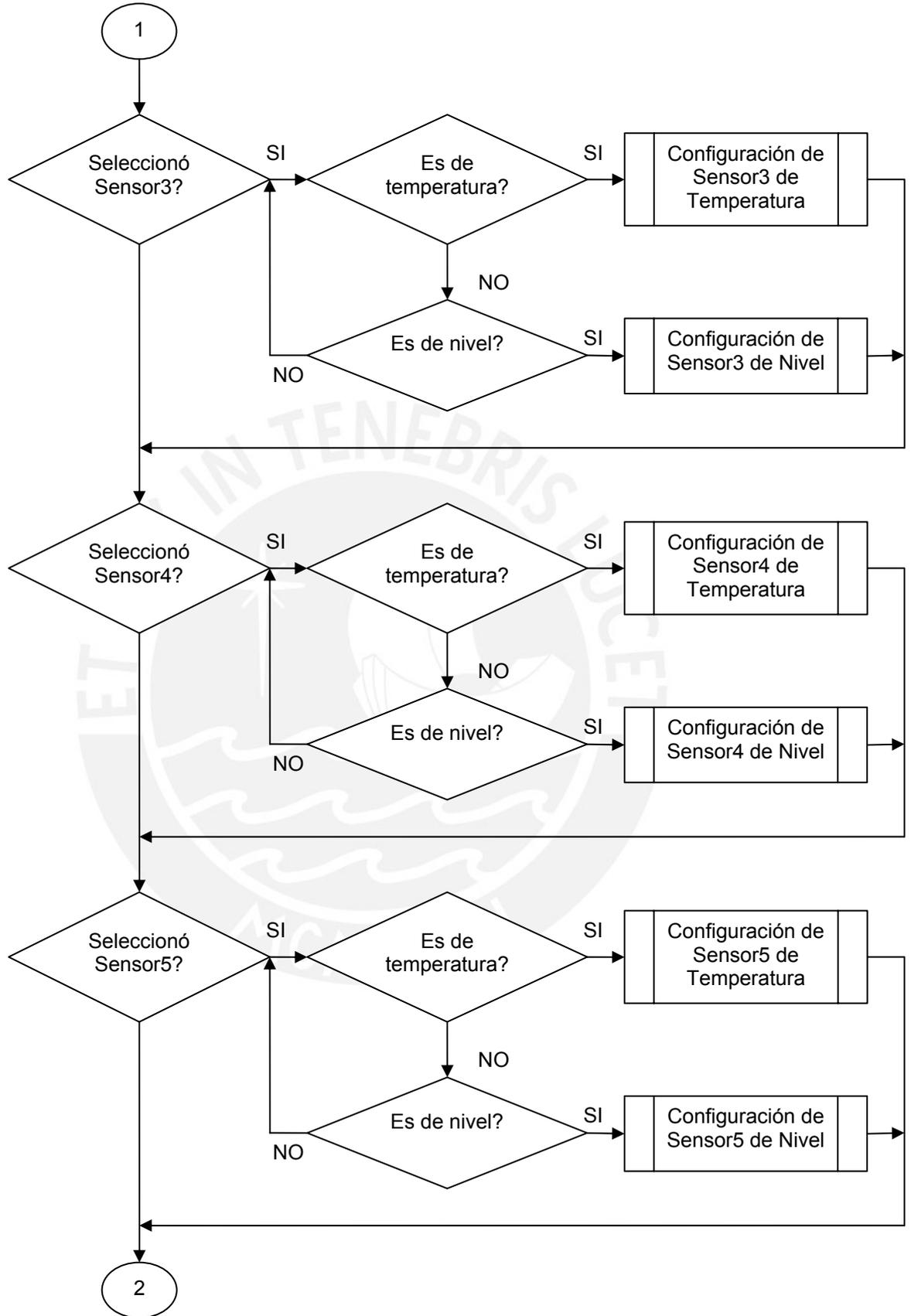
4. DISEÑO, PRUEBAS Y RESULTADOS

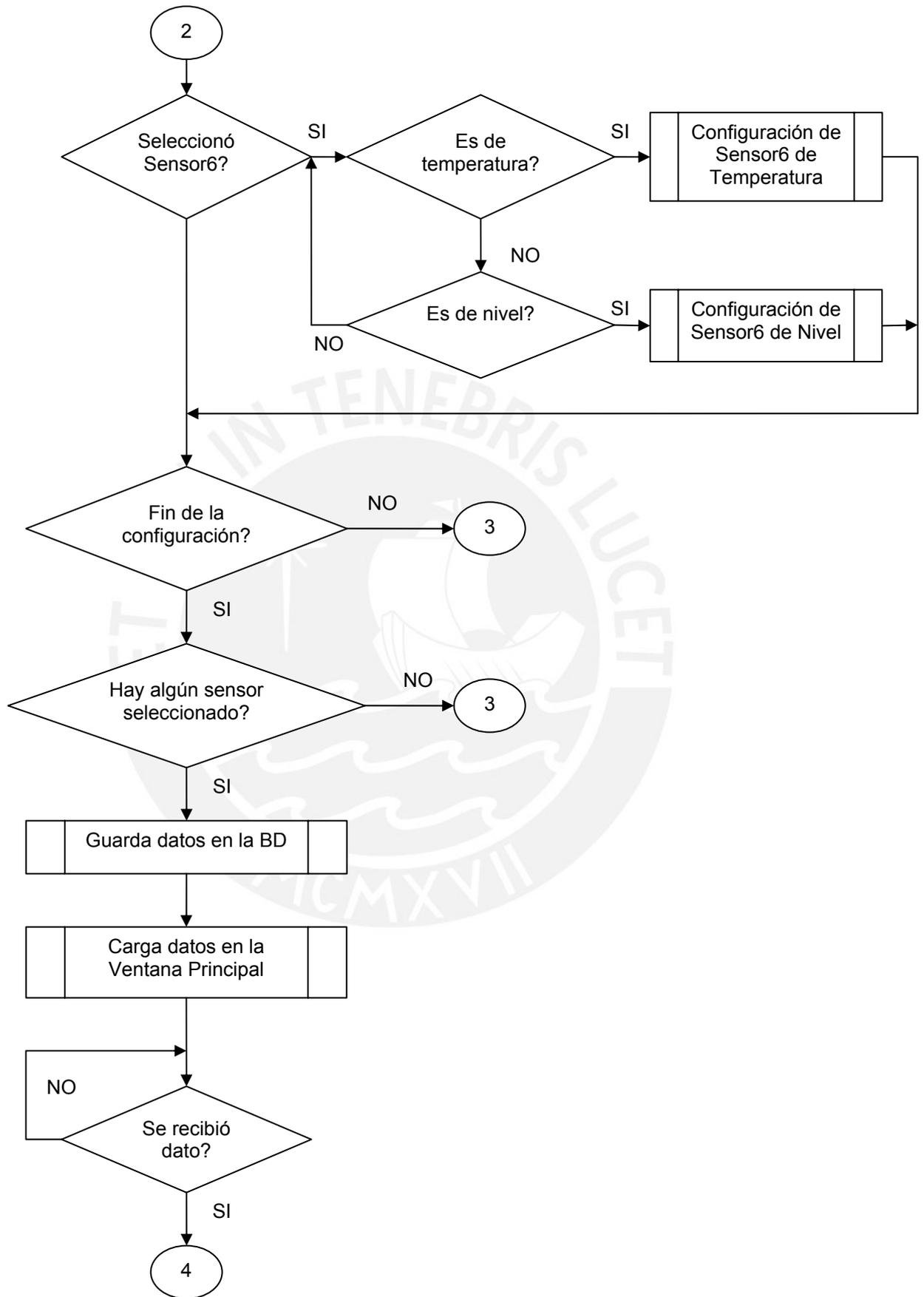
4.1 Diagramas de Flujo y Ventanas

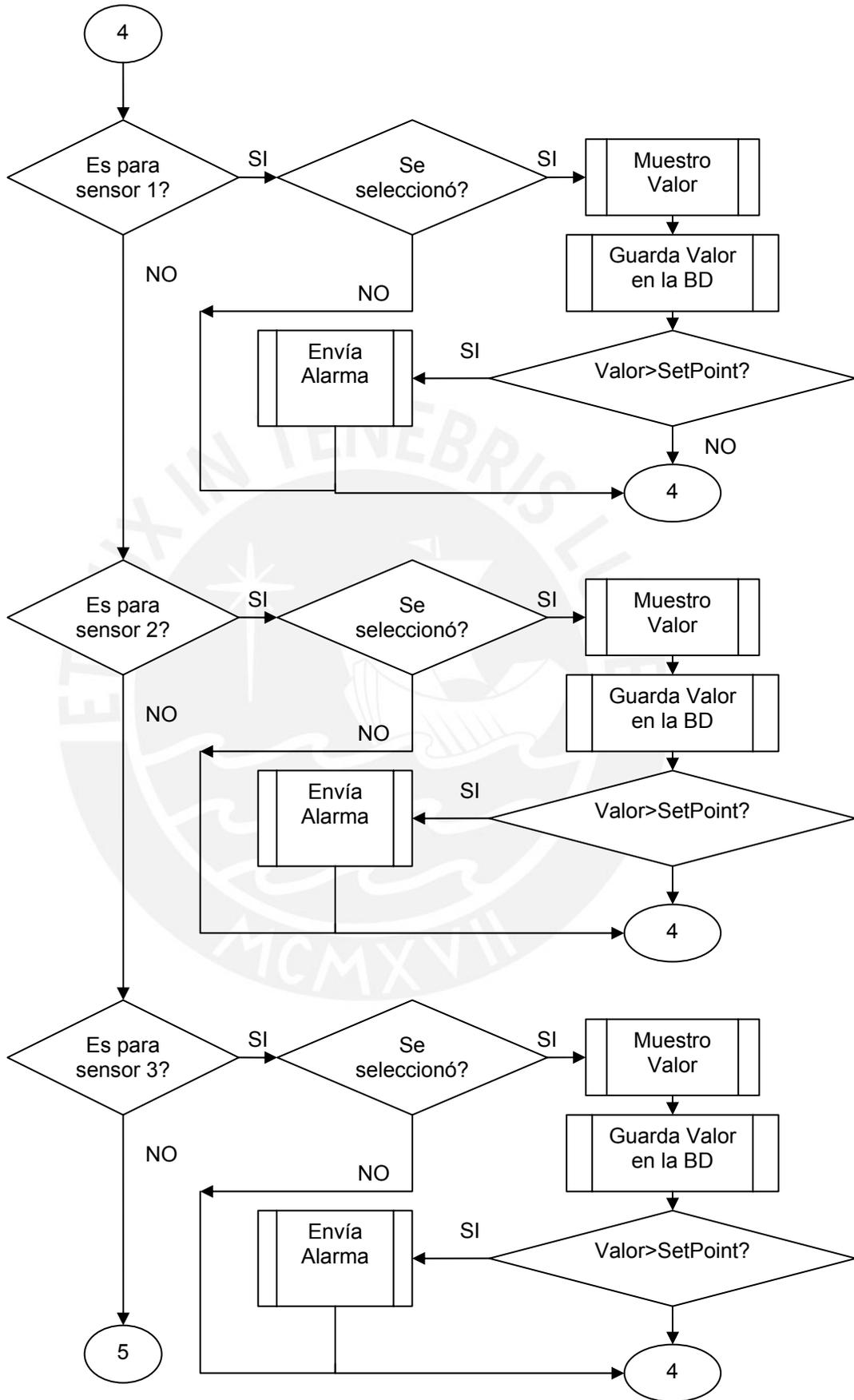
4.1.1 Programa Servidor

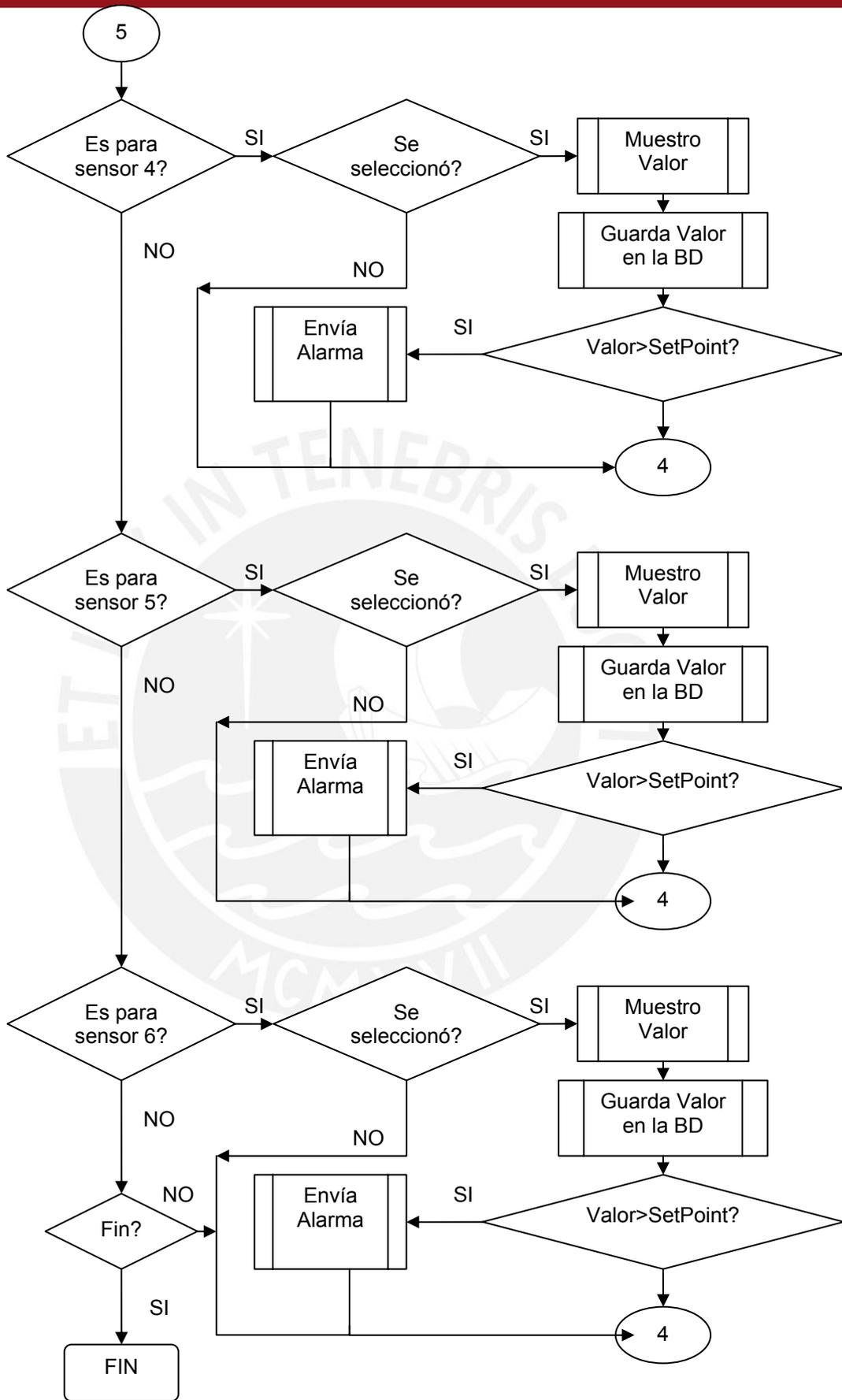
El programa se encarga de la configuración de los tipos de sensores a usar, seis en total, así como la adquisición de datos vía serial provenientes de la tarjeta de acondicionamiento y la visualización gráfica y numérica de los mismos. El usuario en un primer inicio configura los sensores (tipo, rango mínimo, rango máximo, set point, unidades de medida), a continuación envía los valores a la base de datos y procede a visualizar los datos en la pantalla de configuración. Si los valores de cualquiera de los sensores supera el rango del set point el programa envía un mensaje de control a la tarjeta de acondicionamiento para que se accione la alarma que esté conectada a un determinado sensor.











Ventana Inicio de Sesión

Primera ventana que se muestra al usuario al iniciar el programa. Confirma con la base de datos que el nombre de usuario coincida con la contraseña (password) ingresado (Figura 4.1).

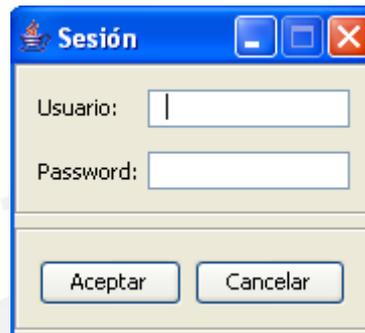


Figura 4.1. Ventana Inicio de Sesión.

Ventana Principal

Ventana principal que se muestra luego de la ventana de inicio de sesión donde se visualiza en el panel de datos el nombre del usuario y la fecha del inicio de la sesión (Figura 4.2).

En el panel de configuración el usuario puede seleccionar entre 6 sensores diferentes a usar y además seleccionar el tipo de sensor, temperatura o nivel. Luego de configurarlo para empezar la toma de datos se procede en apretar el botón ENVIAR, esto también con la finalidad de guardar los valores de configuración en la base de datos.

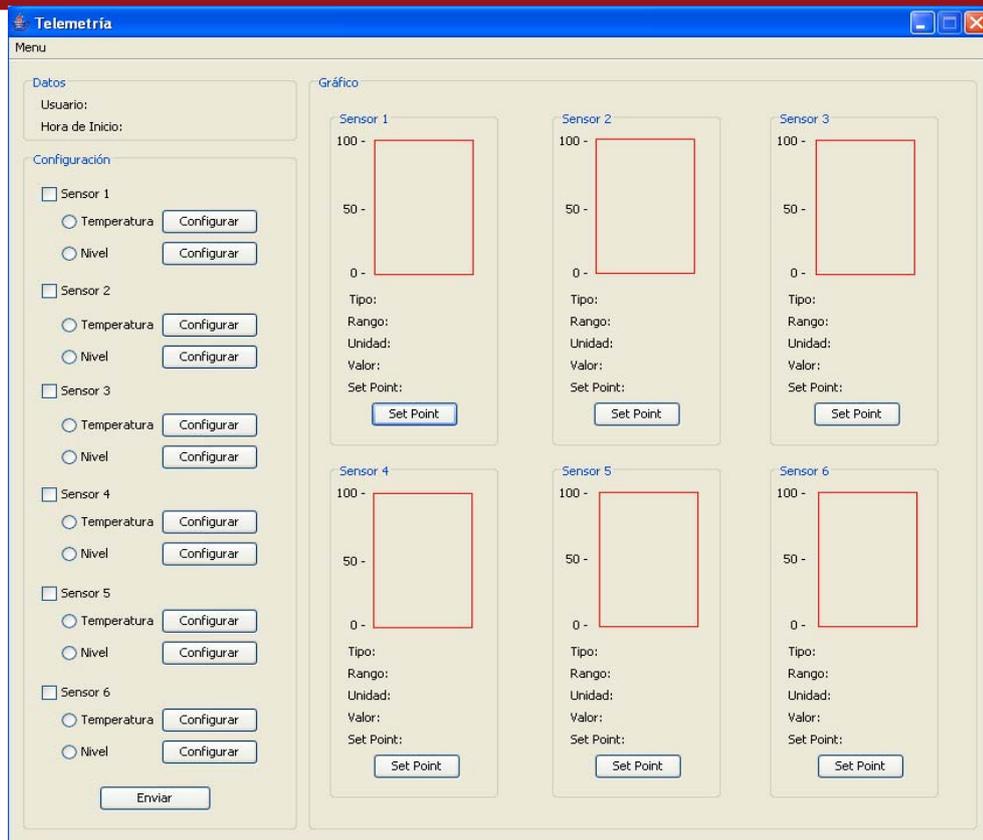


Figura 4.2. Ventana Principal del programa.

Ventana Configuración Sensor de Temperatura

Ventana en la cual se configura el sensor de temperatura (Figura 4.3) con las siguientes restricciones:

- La temperatura mínima que ingresa el usuario debe ser mayor que la temperatura máxima
- El Set Point debe estar dentro del rango de temperatura ingresado.



Figura 4.3. Ventana sensor de temperatura.

Ventana Configuración Sensor de Nivel

Ventana en la cual se configura el sensor de nivel (Figura 4.4) con las siguientes restricciones:

- El valor del nivel mínimo debe ser menor al valor máximo y además mayor a cero.
- El Set Point debe estar dentro del rango de nivel ingresado.

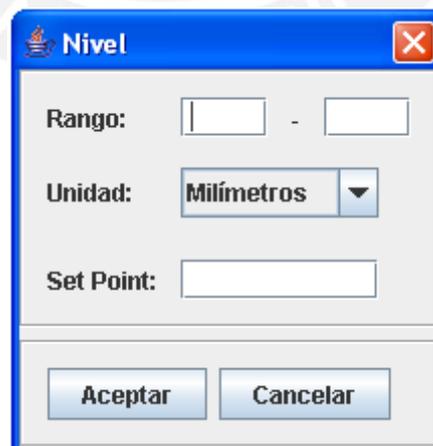


Figura 4.4. Ventana sensor de nivel.

Ventana Set Point

En esta ventana se modifica el valor del Set Point en tiempo real de muestreo, con el botón de SET POINT del panel de Gráfica (Figura 4.5).

Este dato es actualizado en la base de datos.

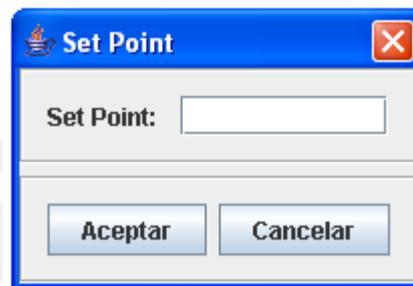
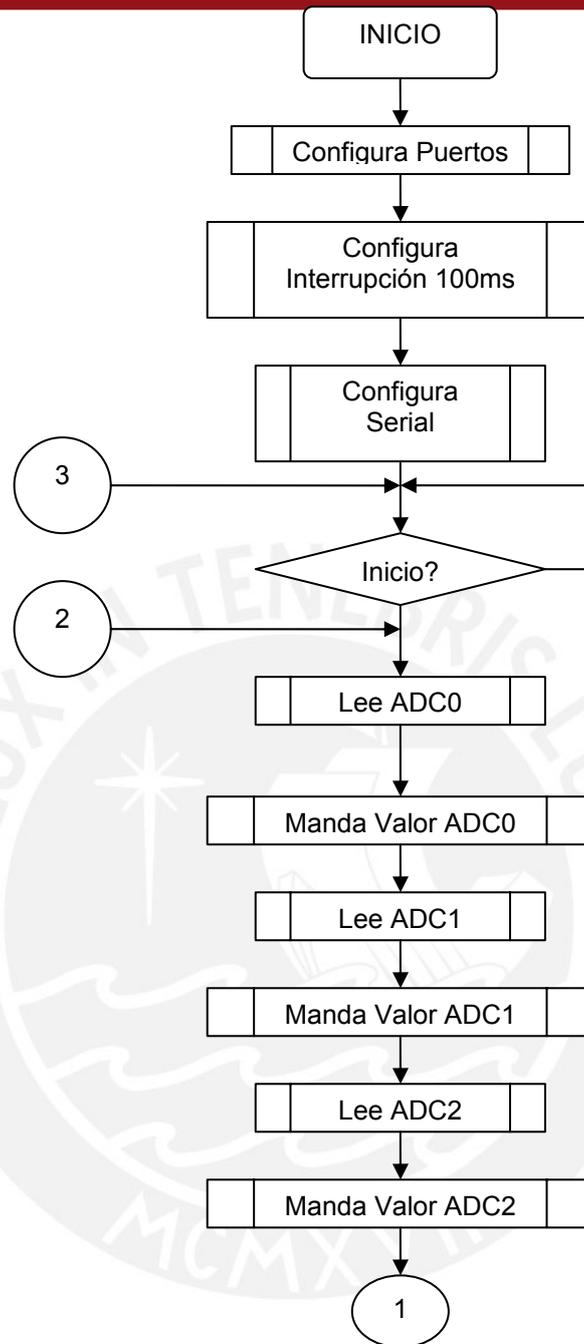
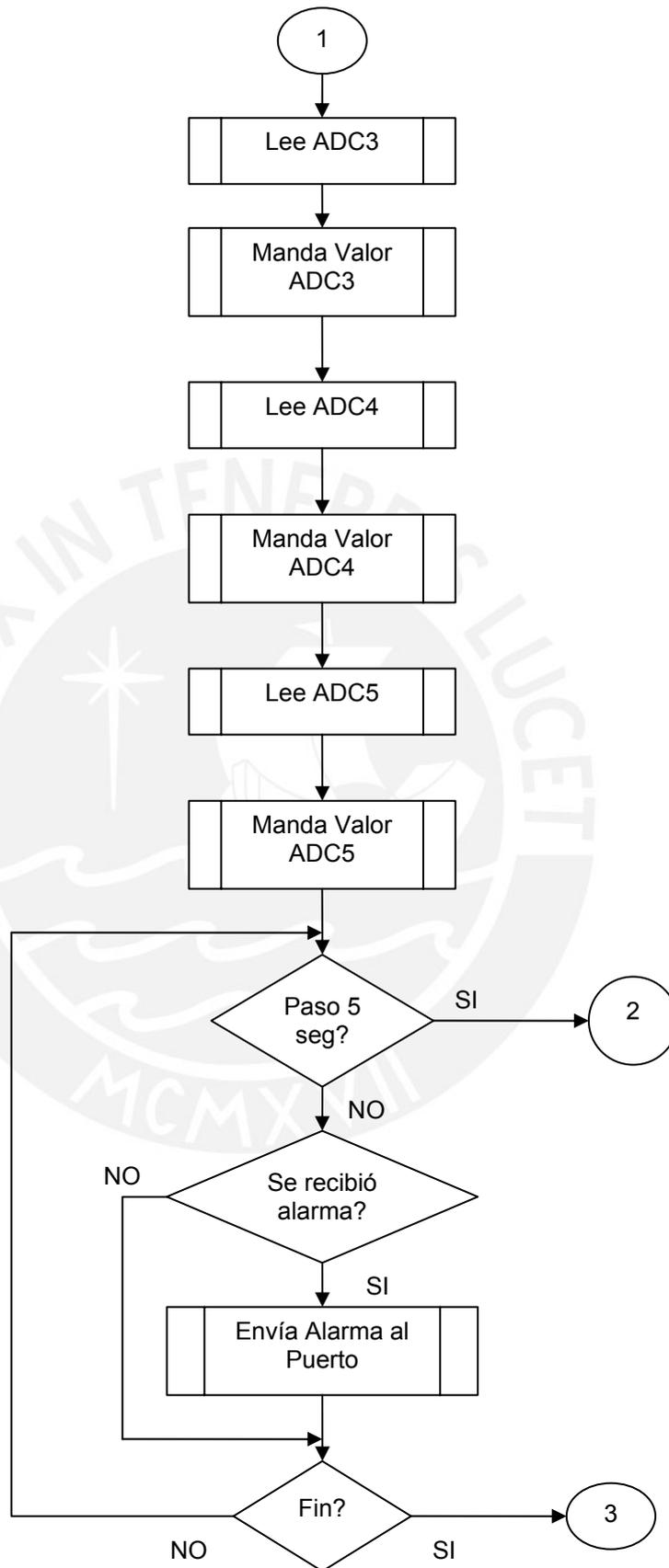


Figura 4.5. Ventana que sirve para modificar el set point.

4.1.2 Programa del Microcontrolador

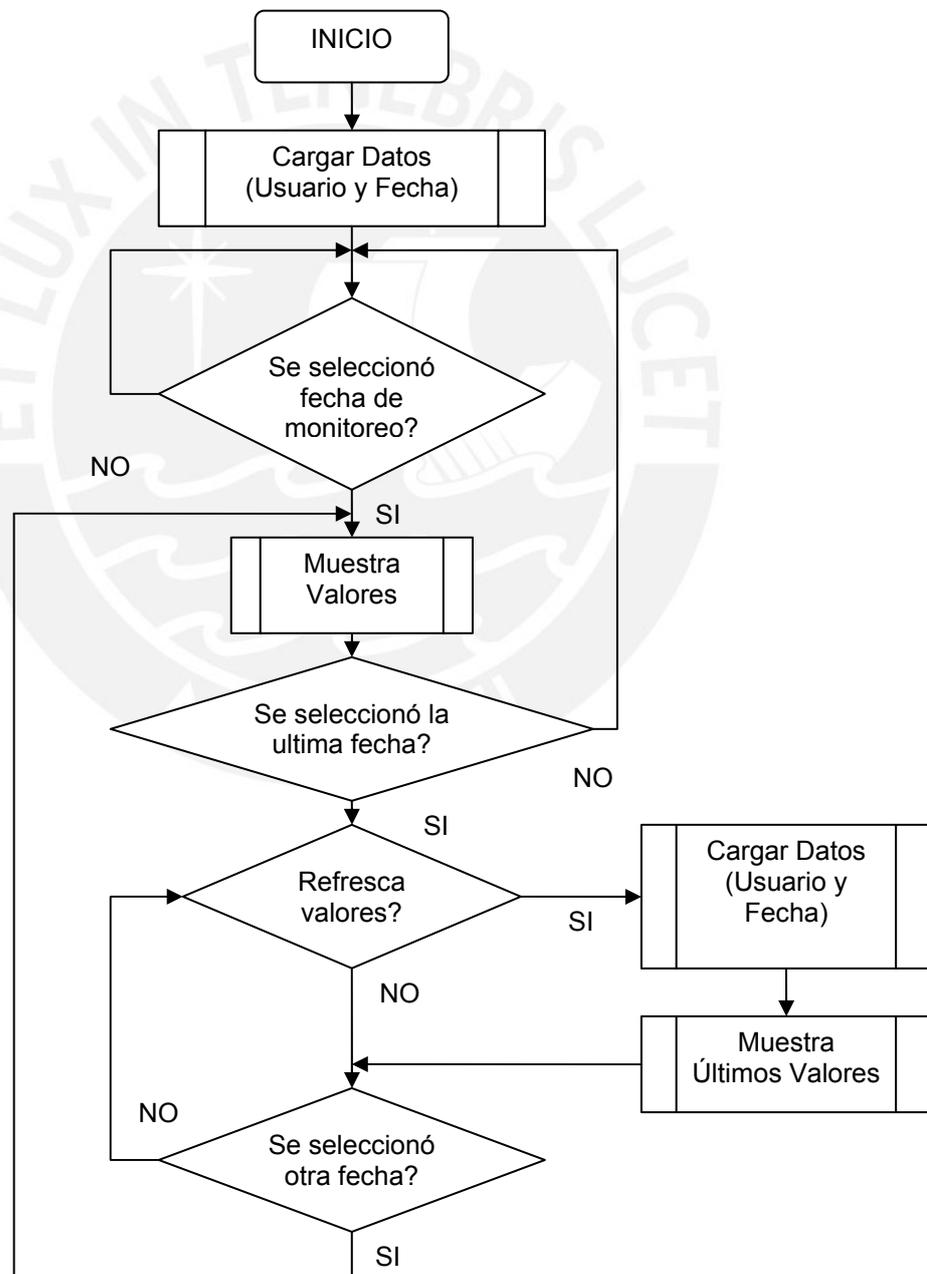
El programa se encarga de digitalizar cada 2 segundos la señal analógica de los ADC internos y enviarlas vía serial al computador principal, así como también mandar por los pines de control las señales de alarma (Ver Anexo 4).





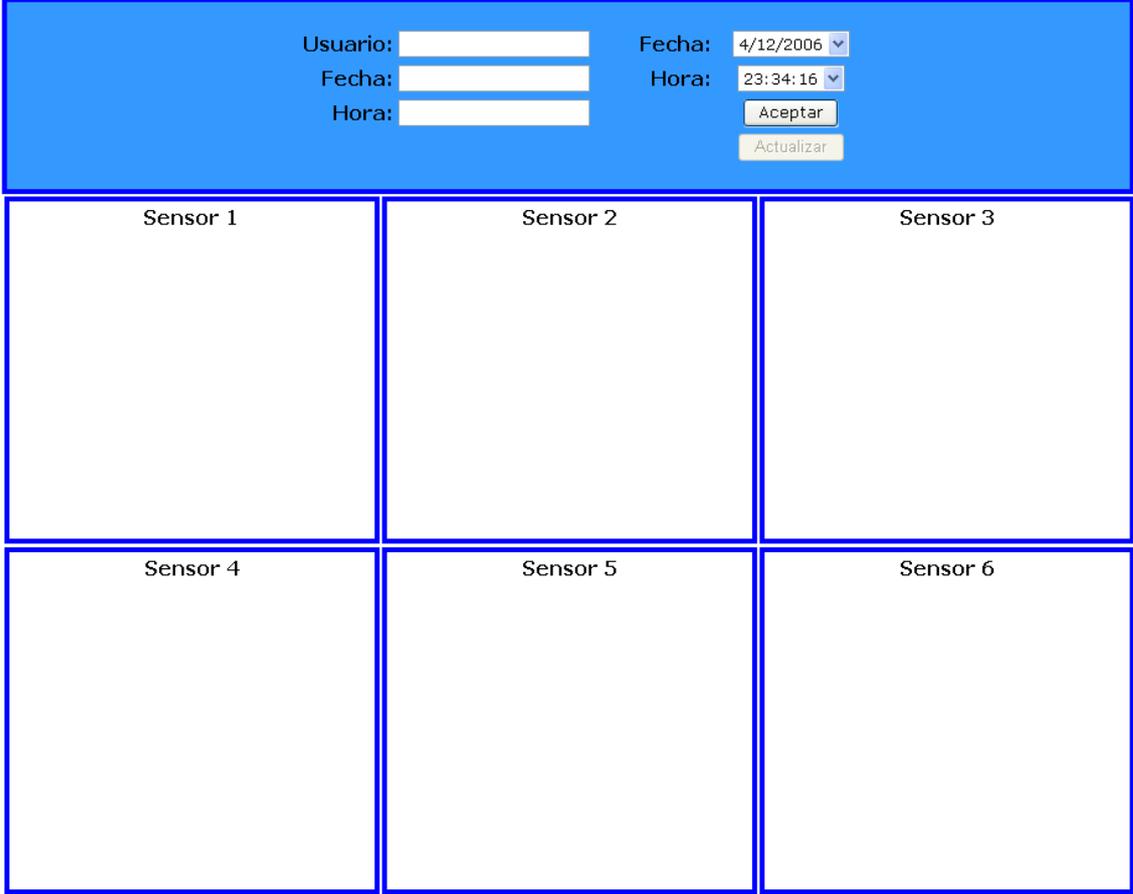
4.1.3 Programa de Aplicación WEB

El programa se encarga de la visualización de los parámetros en tiempo real o guardados en la base de datos con anterioridad. Esto se da seleccionando la fecha y hora del evento que se quiere visualizar, y si es un monitoreo en curso se podrá refrescar los datos para visualizarlos en el momento presente.



Ventana Principal de Aplicación Web

Ventana principal donde el usuario selecciona la fecha y hora de los resultados que desea visualizar (Figura 4.6).



The screenshot shows a web application interface. At the top, there is a blue header bar containing a form. The form has three input fields for 'Usuario', 'Fecha', and 'Hora'. The 'Fecha' field is a dropdown menu showing '4/12/2006'. The 'Hora' field is a dropdown menu showing '23:34:16'. Below these fields are two buttons: 'Aceptar' and 'Actualizar'. Below the header bar is a grid of six empty rectangular boxes, each labeled 'Sensor 1' through 'Sensor 6'.

Figura 4.6. Ventana principal donde se visualizan los datos de los sensores.

4.2 Pruebas y Resultados

4.2.1 Programa Servidor

Las pruebas hechas al software encargado del monitoreo se realizan en dos partes, la primera correspondiente a la validación de los parámetros ingresados por el usuario, así como la validación de las restricciones propias del software, lo que se debe hacer para su correcto uso, los pasos a seguir.

En la primera parte se describe la etapa de inicio de sesión del usuario la cual se valida que los campos USUARIO y PASSWORD no estén en blanco y también la validación del nombre de usuario con la respectiva contraseña ingresada por el usuario, teniéndose en cuenta de que el usuario no ingrese ningún dato (Figura 4.7) o que lo ingrese incorrectamente (Figura 4.8).

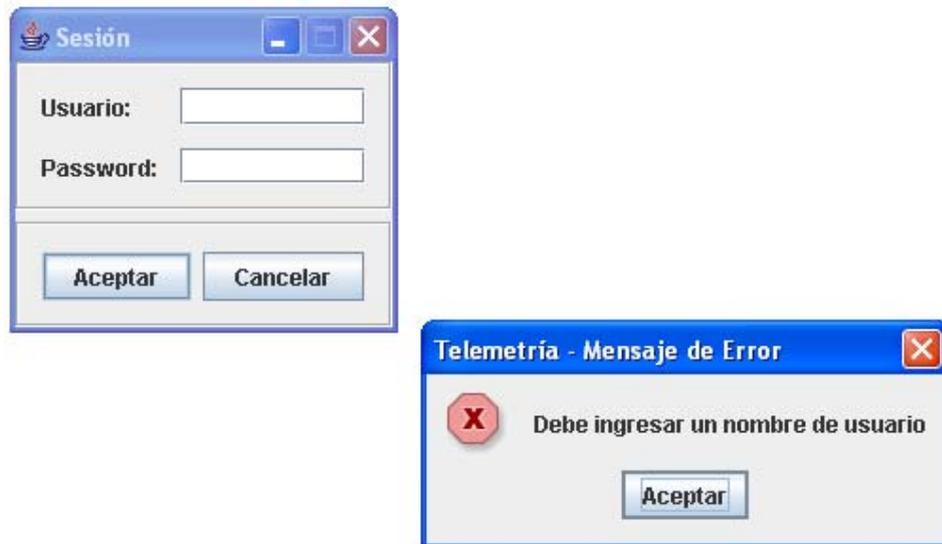


Figura 4.7. Inicio de sesión sin ingresar nombre de usuario ni contraseña.

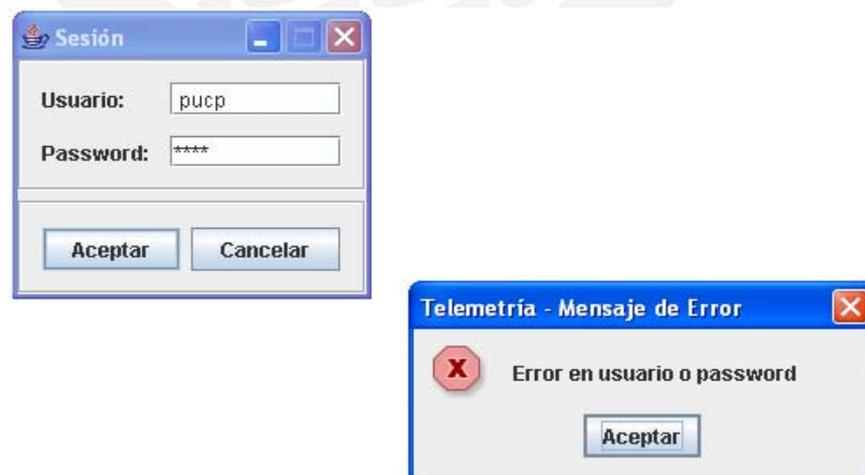


Figura 4.8. Inicio de sesión con una contraseña no correspondiente al nombre de usuario o un nombre de usuario no válido.

Estos datos anteriores son verificados de la base de datos, la cual tiene guardados los nombres de los usuarios y su respectiva contraseña (Figura 4.9).

Table - dbo.tbl_configuracion		Table - dbo.tbl_usuario		Summary
	Nombre	Password	Estado	idUsuario
	javier	javier	0	1
	carlos	carlos	0	2
▶*	NULL	NULL	NULL	NULL

Figura 4.9. Datos de los usuarios en la base de datos.

A continuación una vez ingresado el nombre de usuario y la contraseña válida se procede a cargar el programa principal en el cual se prosigue con una etapa de configuración de sensores las cuales tienen las siguientes restricciones al usuario:

- Antes de apretar el botón ENVIAR, la cual envía los valores de configuración a la base de datos se debe configurar por lo menos un sensor (Figura 4.10).

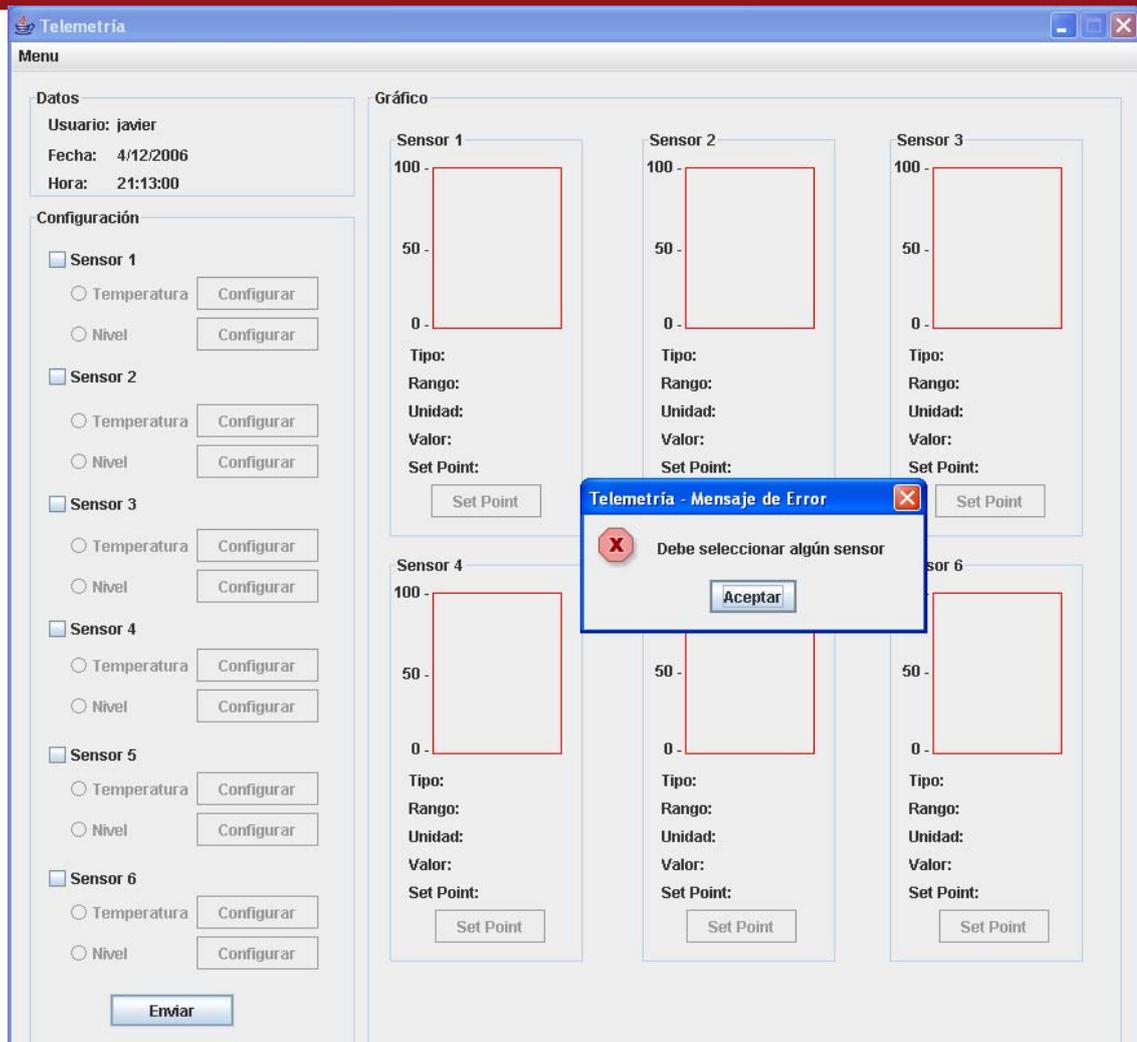


Figura 4.10. Mensaje de Error al no seleccionar ningún sensor.

Como segundo paso de las pruebas, se procede a comprobar que los datos tanto de configuración como de valores recibidos sean correctamente almacenados. Para esto se procederá a configurar tres de los seis sensores como se muestra en la Tabla 4.1.

Número de Sensor	Tipo	Rango Mínimo	Rango Máximo	Set Point	Unidades
1	Temperatura	10	80	40	Centígrados
3	Nivel	5	30	20	Metros
5	Temperatura	30	150	100	Centígrados

Tabla 4.1. Datos de la configuración de los sensores.

Como los datos provenientes de los sensores van a unos comparadores análogos digitales (ADC) de una resolución de 8 bits, los valores que llegan serialmente al computador son valores digitales entre 0 y 255. Además se tiene en cuenta que para los sensores de tipo de temperatura las señales ingresadas a los ADC's son señales de voltajes en los rangos de 0 – 5 VDC, ya que dichos sensores utilizan esta señal estándar, sin embargo los sensores de nivel industrial trabajan con señales en el rango de 4 – 20 mA, por lo que los niveles de voltaje de entrada a los ADC's están en el rango de 1 – 5 VDC. Esto hace que sea necesaria una corrección del cero para los sensores de tipo nivel, ya que para éstos el rango mínimo corresponderá el valor digital de 51.

Así se emplea dos fórmulas para el cálculo de los valores en las unidades de temperatura o nivel.

- Sensor de Temperatura

$$X = \text{RangoMínimo} + \frac{\text{ValorDigital} \times (\text{RangoMáximo} - \text{RangoMínimo})}{255}$$

- Sensor de Nivel

$$X = \text{RangoMínimo} + \frac{(\text{ValorDigital} - 51) \times (\text{RangoMáximo} - \text{RangoMínimo})}{204}$$

X: valor dentro del rango de temperatura o nivel.

ValorDigital: valor digital recibido serialmente (0 – 255).

Las pruebas realizadas en esta etapa corresponden a la visualización, de los valores digitales enviados por el microcontrolador, en valores dentro del rango del sensor escogido. Para esto se procedió a enviar valores digitales en el rango de 0 – 255 mediante el programa *hyperterminal* (Figura 4.11) de una computadora con el correspondiente protocolo de envío.

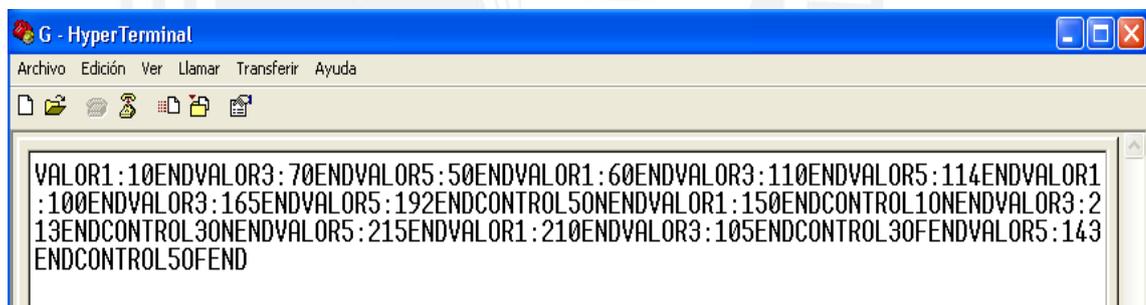


Figura 4.11. Protocolo enviado mediante el *hyperterminal*.

El protocolo de envío de valores es el siguiente:

VALOR[X]:[ValorDigital]END

El protocolo de envío de señales de control es la siguiente:

CONTROL[X]ONEND

CONTROL[X]OFEND

Donde: X es el número de sensor (1 – 6)

Valor Digital: número digital del ADC (0 – 255)

Se aprecia que cuando el valor sobrepasa al Set Point envía una señal de control de ON y cuando regresa a su estado normal (por debajo del Set Point) envía una señal de OFF.

Así mismo se listan los valores de voltaje correspondientes a los valores digitales enviados (Tabla 4.2).

Sensor	Valor Digital	Valor del Sensor (teórico)
1	10	12.75
	60	26.47
	100	37.45
	150	51.18
	210	67.65
3	70	7.33
	110	12.23
	165	18.97
	213	24.85
	105	11.62

5	50	53.53
	114	83.65
	192	120.35
	215	131.18
	143	97.29

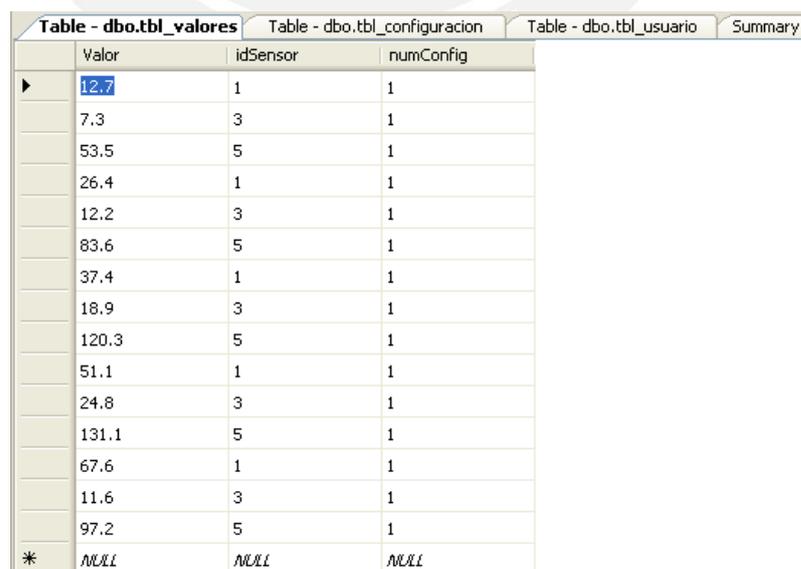
Tabla 4.2. Valores digitales y su correspondiente valor de voltaje.

También se comprobó que los datos tanto de configuración (Figura 4.13) como los valores de voltaje (Figura 4.14) fueron guardados en la base de datos.



Tipo	RangoMinimo	RangoMaximo	Unidad	SetPoint	idSensor	numConfig	Fecha	Hora	idUsuario
Temperatura	10	80	Centigrados	40	1	1	4/12/2006	23:34:16	1
Nivel	5	30	Centímetros	20	3	1	4/12/2006	23:34:16	1
Temperatura	30	150	Centigrados	100	5	1	4/12/2006	23:34:16	1
* NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figura 4.13. Tabla de datos de configuración en la base de datos.



Valor	idSensor	numConfig
12.7	1	1
7.3	3	1
53.5	5	1
26.4	1	1
12.2	3	1
83.6	5	1
37.4	1	1
18.9	3	1
120.3	5	1
51.1	1	1
24.8	3	1
131.1	5	1
67.6	1	1
11.6	3	1
97.2	5	1
* NULL	NULL	NULL

Figura 4.14. Tabla de valores en la base de datos.

Se obtuvo así una pantalla final con los últimos valores enviados como se muestra en la Figura 4.15.

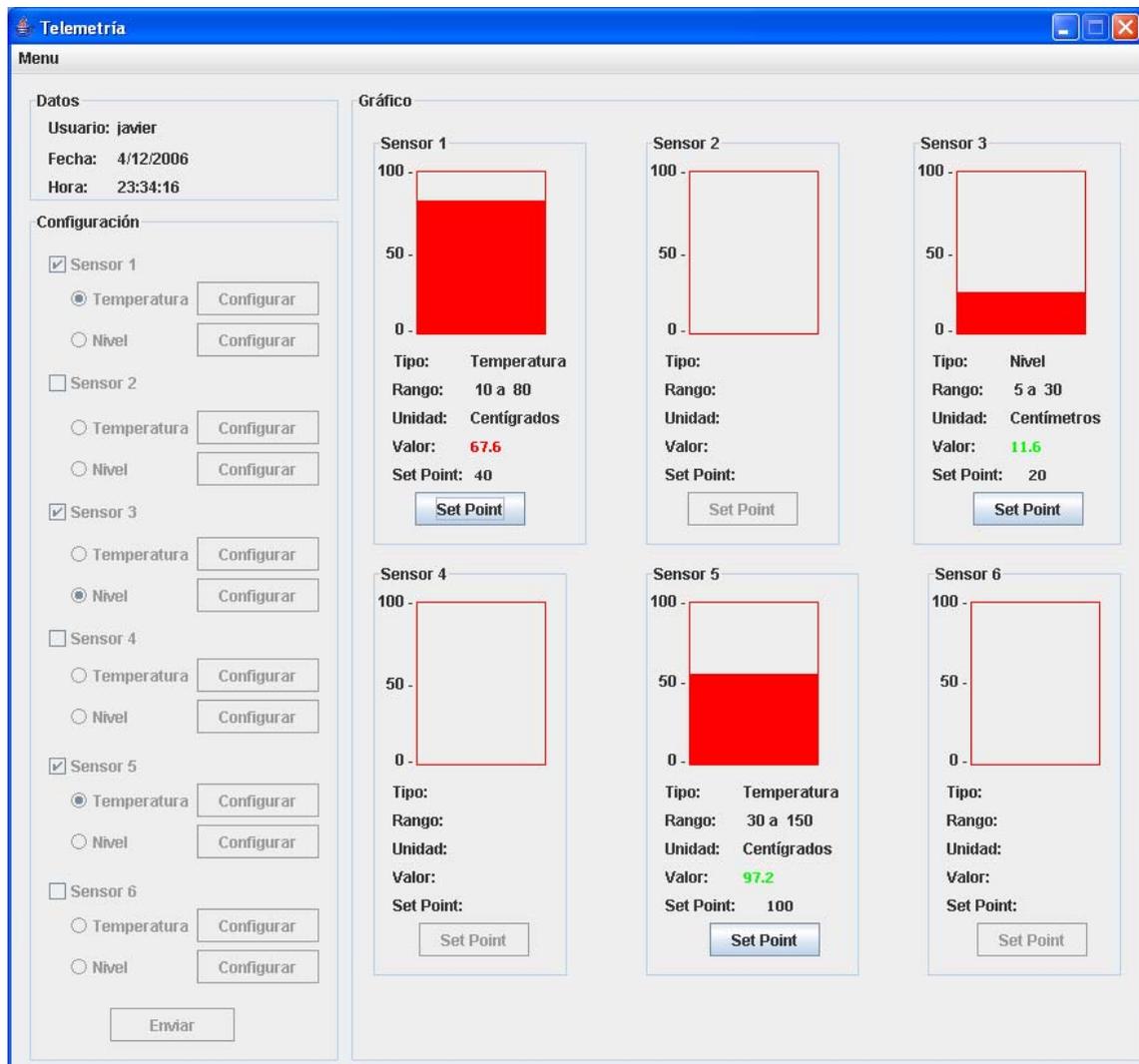


Figura 4.15. Pantalla final con los valores enviados.

4.2.2 Programa Microcontrolador

Para las pruebas del programa del microcontrolador, como éste es el encargado de acondicionar la señal proveniente de los sensores utilizando los comparadores análogo digital interno, se mandó señales de voltaje 0 – 5 VDC, para probar que los valores digitales sean los esperados según la

resolución del ADC (8 bits). A continuación se muestra en la Tabla 4.3 los valores de voltaje ingresados al ADC y su respectivo valor digital dado por el microcontrolador.

Voltaje de Entrada (V)	Valor Digital Teórico	Valor Digital Real	Error (V)
0	0	0	0
0.5	26	26	0
1	51	51	0
1.5	77	78	+0.0196
2	102	100	-0.0392
2.5	128	128	0
3	153	154	+0.0196
3.5	179	179	0
4	204	201	0
4.5	230	232	+0.0588
5	255	253	-0.0392

Tabla 4.3. Valores de voltaje y digitales provenientes del módulo de acondicionamiento.

Resolución = 0.0196 V

Para las pruebas, se uso un multímetro FLUKE con una resolución de 2 decimales de voltaje.

4.2.3 Programa de Aplicación WEB

Para el programa WEB realizado, las pruebas consistieron en cargar los valores antes guardados en la base de datos por el programa encargado del monitoreo.

Al seleccionar la fecha y luego la hora correspondiente a los valores que se quieren visualizar, se procede a apretar el botón ACEPTAR para cargar los datos como se muestra en la Figura 4.16.

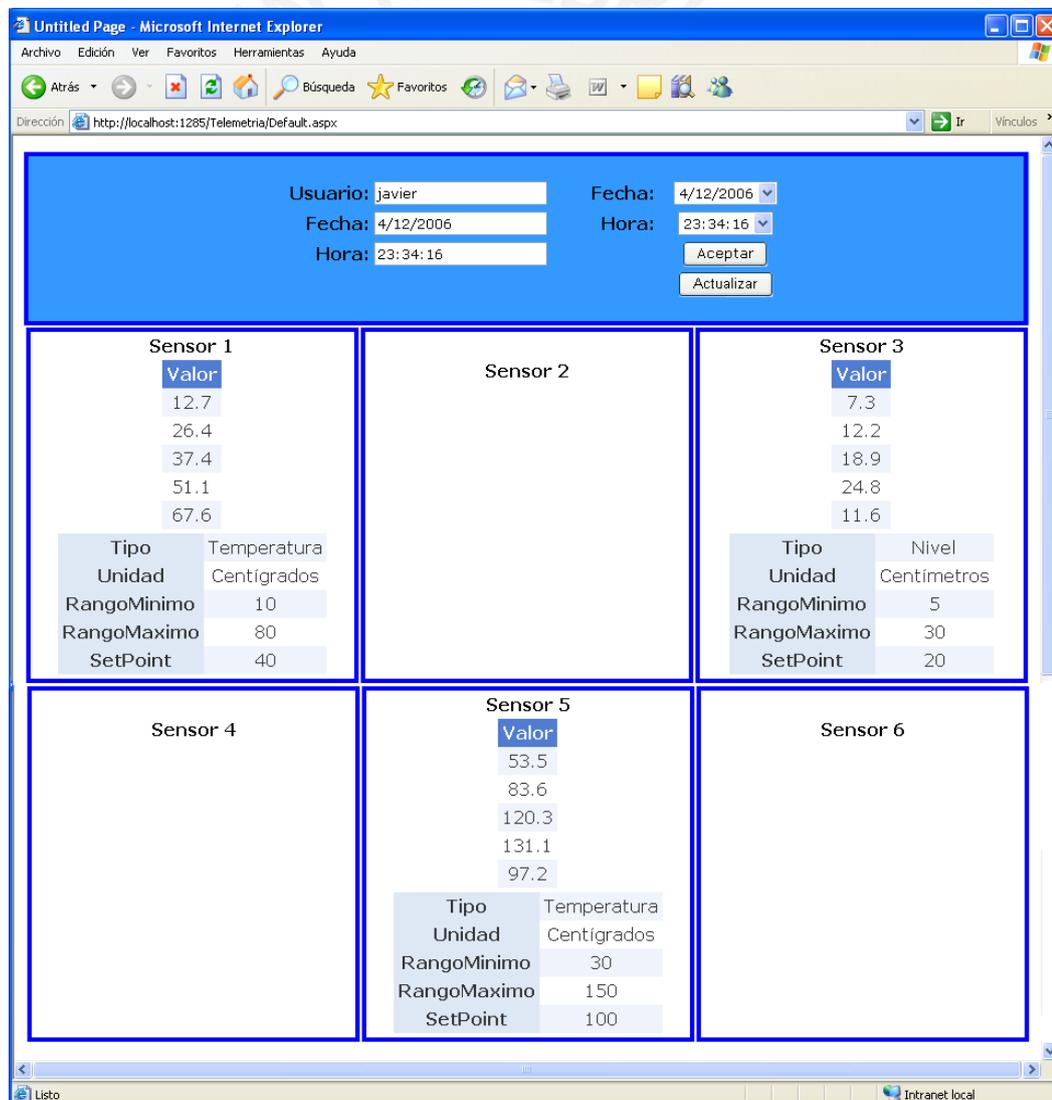


Figura 4.16. Página Web con los valores cargados de la base de datos.

Si la fecha y hora seleccionadas corresponden a una proceso de monitoreo en curso, si se aprieta el botón ACTUALIZAR, carga el o los últimos valores guardados en la base de datos a fin de que el usuario pueda visualizarlos al momento como se muestra en la Figura 4.17.

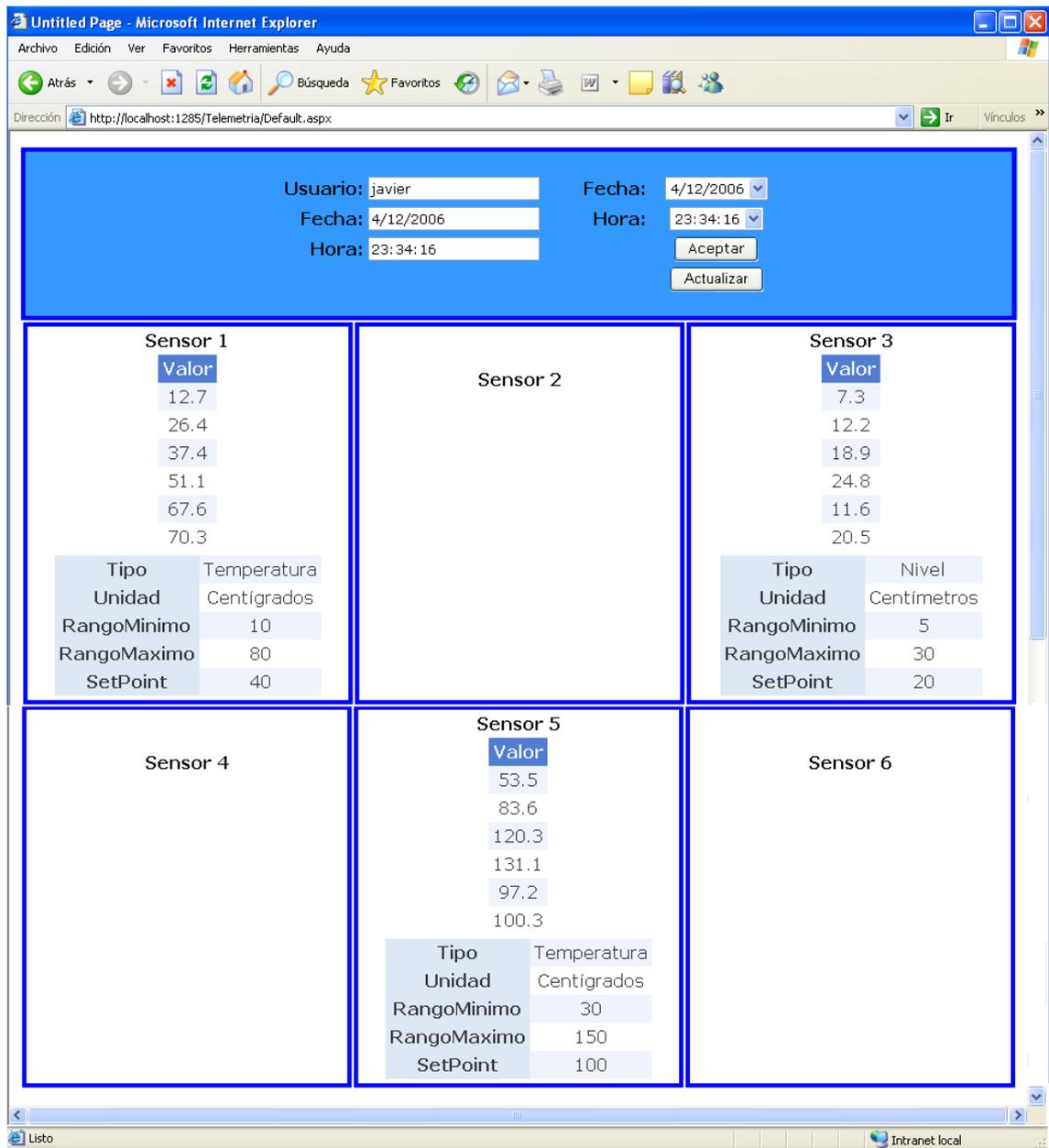


Figura 4.17. Página Web con los valores actualizados.

Se puede ver como los nuevos valores de los sensores 1 (70.3), 2 (20.5) y 3 (100.3) son refrescados en la pantalla, según lo actualizado en la base de datos (Figura 4.18).

Table - dbo.tbl_valores		Summary	
	Valor	idSensor	numConfig
	12.7	1	1
	7.3	3	1
	53.5	5	1
	26.4	1	1
	12.2	3	1
	83.6	5	1
	37.4	1	1
	18.9	3	1
	120.3	5	1
	51.1	1	1
	24.8	3	1
	131.1	5	1
	67.6	1	1
	11.6	3	1
	97.2	5	1
	70.3	1	1
▶	20.5	3	1
	100.3	5	1
*	NULL	NULL	NULL

Figura 4.18. Valores de voltaje actualizados (rojo) guardados en la base de datos.

4.3 Diseño de la Tarjeta de Acondicionamiento

La tarjeta de acondicionamiento consta básicamente de un microcontrolador ATmega8L del cual se hace uso de seis entradas para los ADC (PC0...PC5), así como 6 salidas digitales para las señales de alarma (PB0...PB5). Además se añadió las conexiones necesarias para la programación del microcontrolador (ISP), así como un MAX232 para la comunicación serial

con una salida de conector DB9 macho, y un circuito regulador de voltaje de 5V. El diagrama esquemático es como se muestra en la Figura 4.19.

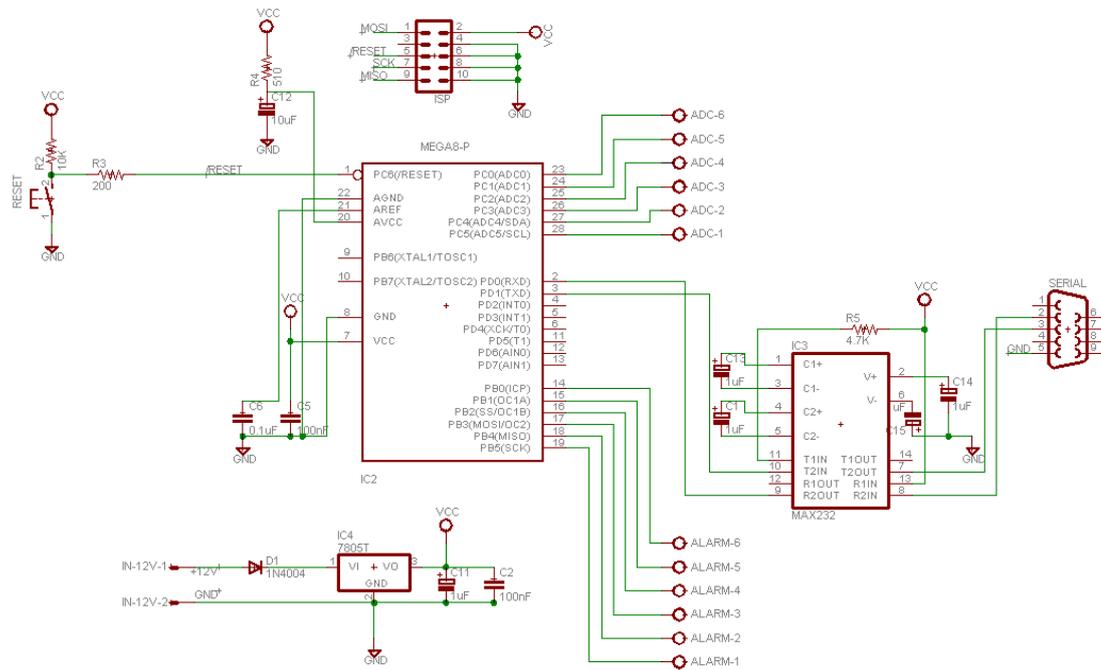


Figura 4.19. Diagrama Esquemático de la Tarjeta de Acondicionamiento.

4.4 Costos del Diseño

Para los costos del diseño, se tomaron en cuenta el hardware, software y la inversión de horas hombre para la programación del respectivo software. Los costos se detallan en la Tabla 4.4.

ITEM	DESCRIPCIÓN	UNIDAD	CANTIDAD	PRECIO UNITARIO (\$)	PRECIO TOTAL (\$)
Hardware					
1	Sensor de Nivel	EA	4	200.00	800.00
2	Sensor de Temperatura PT-100	EA	2	100.00	200.00
3	Cable AWG 18 apantallado con shield. Para el sensor de nivel.	Mt.	30	1.50	45.00
4	Cable para PT-100 (PVC).	Mt.	14.5	1.30	18.85
5	Cable de cobre.	Mt.	10	0.50	5.00
6	Circuito de acondicionamiento de señal	EA	1	50.00	50.00
7	Canaleta (1.8 mts)	EA	7	0.60	4.20
8	Caja de paso	EA	1	10.00	10.00
Software					
7	Microsoft Visual WEB developer 2005 Express Edition.	EA	1	43.44	43.44
8	Dominio de la dirección WEB (*)	EA	1	10.00	10.00
9	WEB Hosting (*)	EA	1	20.00	20.00
Programación					
10	Elaboración de software.	Hr.	120	3.00	360.00
				TOTAL	1566.49

Tabla 4.4. Detalle de costos del diseño.

(*) Pago anual

Se toma en cuenta que la plataforma donde se va a ejecutar los programas es sobre el sistema operativo LINUX por lo que no se toma en cuenta un costo por licencia de Microsoft Windows XP.

CONCLUSIONES

Luego de las pruebas realizadas, la comunicación serial RS-232 satisface las necesidades de comunicación entre dispositivos, en este caso módulo de acondicionamiento y computador terminal, dando también un fácil manejo de la transmisión de datos y sobre todo que este tipo de comunicación es soportado por la mayoría de dispositivos (hardware) y lenguajes de programación (software).

Las pruebas con la base de datos también se registraron correctamente, guardando de manera exacta los valores recibidos en el computador principal, también registrando el ingreso del usuario y almacenando los datos dentro de su historial.

Los resultados obtenidos de la base de datos, se cargaron de manera exitosa en el programa WEB, registrando solo los sensores configurados y cargando los datos en los respectivos parámetros (valor máximo, valor mínimo, set point, etc.).

Entonces el sistema en conjunto no solo toma valores de los sensores sino que acondiciona la señal para su posterior visualización ya sea en el lugar mismo de trabajo o en un lugar remoto vía Internet (WEB).

RECOMENDACIONES

En cuanto a las especificaciones del hardware usar se debe tener en cuenta las señales estándares de entradas para los conversores análogo digital del microcontrolador, las cuales deben ser de 0 – 5 VDC para el caso de sensores de temperatura y de 1 – 5 VDC para el caso de sensores de nivel.

En cuanto a las salidas digitales que proporciona la tarjeta de acondicionamiento, se debe tener en cuenta que la corriente máxima dada por el microcontrolador puede ser como máximo de 20 mA para un adecuado funcionamiento.

Para un funcionamiento mas completo, en cuanto al proceso en sí de tostado de granos de kiwicha, se debe desarrollar un proceso de control a fin de no solo tener una visualización de los parámetros críticos del proceso sino también poder controlarlos de manera automática, esto para tener un menor margen de error en el proceso y además de optimizar el mismo. Este proceso de control también debe ser implementado tanto en el servidor como en la aplicación Web, ya que si se requiere cambiar algún parámetro, estando el encargado en un lugar distante a la planta de proceso, lo pueda hacer sin la necesidad de ir al computador principal.

Además de lo antes mencionado una mejora tecnológica a desarrollar e implementar es la de una comunicación inalámbrica entre los sensores y el modulo de acondicionamiento, facilitando esto la distribución de los sensores en los equipos.

BIBLIOGRAFÍA

- [1] CISCO SYSTEM, Inc. y CISCO NETWORKING ACADEMY PROGRAM

2004 Academia de Networking de Cisco Systems Guía del primer año CCNA 1 y 2. Tercera Edición
- [2] LEE, Thomas y DAVIES Joseph

2000 Microsoft Windows 2000 TCP/IP Protocolos y Servicios.
Referencia técnica
- [3] HUIDOBRO, José Manuel

2001 Redes y Servicios de Telecomunicaciones
- [4] PALLÁS, Ramón

1998 Sensores y acondicionadores de señal. 3° edición
- [5] CARRANZA, Raymundo

2001 Automatización Tópicos de Instrumentación y Control
- [6] Casadomo Soluciones S.L.

2006 Arquitectura y Estructura de Dispositivos y Redes [en línea].
2006 [consultado el 2006-04-15]. <http://www.construible.es/noticiasDetalle.aspx?c=50&m=61&idm=61&pat=59&n2=59>

- [7] ROMÁN, Isabel
- 2005 Estructura de las Redes de Telecomunicaciones [en línea].
2006 [consultado el 2006-04-20].
<http://trajano.us.es/~isabel/publicaciones/Tema3.pdf>
- [8] Real Academia Española
- <http://www.rae.es/>
- [9] Asociación Mexicana de Amaranto
- 2003 Amarantum: Asociación Mexicana de Amaranto [en línea].
[consultado el 2006-04-20].
<http://www.amaranta.com.mx/elamaranto/secAmaranto.html>
- [10] IBARRA, Silvia
- UNESCO – Amaranto [en línea]. [consultado el 2006-04-25].
<http://www.elportaldemexico.com/arteculinario/pueblodemaiz-unesco.htm>
- [11] Rockwell Automation [consultado el 2007-02-27].
http://www.ab.com/catalogs/C114-CA001A-ES-P/1a_photo.pdf
- [12] ARIAN Control & Instrumentation [consultado el 2007-03-01]
<http://www.arian.cl/downloads/nt-004.pdf>

ESPOSITO, Anthony

1980 Fluid Power with Applications

Centro de Investigación, Educación y desarrollo [en línea]. [consultado el 2006-04-25]. <http://www.ciedperu.org/productos/kiwicha.htm>

ATMEL

http://www.atmel.com/dyn/resources/prod_documents/doc2486.pdf



ANEXOS

Anexo 1

Pt 100 ohms	0	1	2	3	4	5	6	7	8	9
-190	22.78	22.35	21.93	21.50	21.08	20.66	20.23	19.81	19.38	18.96
-180	27.01	26.59	26.17	25.74	25.32	24.90	24.47	24.05	23.63	23.20
-170	31.24	30.81	30.39	29.97	29.55	29.13	28.70	28.28	27.86	27.44
-160	35.45	35.03	34.61	34.19	33.77	33.34	32.92	32.50	32.08	31.66
-150	39.65	39.23	38.81	38.39	37.97	37.55	37.13	36.71	36.29	35.87
-140	43.78	43.37	42.96	42.54	42.13	41.72	41.30	40.89	40.48	40.06
-130	47.90	47.49	47.08	46.67	46.26	45.85	45.43	45.02	44.61	44.20
-120	52.01	51.60	51.19	50.78	50.37	49.96	49.55	49.14	48.73	48.32
-110	56.11	55.70	55.29	54.88	54.48	54.07	53.66	53.25	52.84	52.43
-100	60.20	59.79	59.38	58.98	58.57	58.16	57.75	57.34	56.93	56.52
-90	64.23	63.83	63.43	63.02	62.62	62.22	61.81	61.41	61.01	60.60
-80	68.25	67.85	67.45	67.05	66.65	66.25	65.84	65.44	65.04	64.64
-70	72.26	71.86	71.46	71.06	70.66	70.26	69.86	69.46	69.06	68.66
-60	76.26	75.86	75.46	75.06	74.67	74.27	73.87	73.47	73.07	72.67
-50	80.25	79.85	79.45	79.06	78.66	78.26	77.86	77.46	77.06	76.66
-40	84.22	83.83	83.43	83.03	82.64	82.24	81.84	81.44	81.05	80.65
-30	88.18	87.79	87.39	87.00	86.60	86.21	85.81	85.41	85.02	84.62
-20	92.13	91.74	91.35	90.95	90.56	90.16	89.77	89.37	88.98	88.58
-10	96.07	95.68	95.29	94.89	94.50	94.11	93.71	93.32	92.92	92.53
0	100.00	99.61	99.22	98.82	98.43	98.04	97.65	97.25	96.86	96.47
0	100.00	100.39	100.78	101.17	101.56	101.95	102.34	102.73	103.12	103.51
10	103.90	104.29	104.68	105.07	105.46	105.85	106.24	106.63	107.02	107.41
20	107.79	108.18	108.57	108.96	109.35	109.74	110.12	110.51	110.90	111.29
30	111.67	112.06	112.45	112.84	113.22	113.61	114.00	114.38	114.77	115.16
40	115.54	115.93	116.32	116.70	117.09	117.47	117.86	118.24	118.63	119.01
50	119.40	119.78	120.17	120.55	120.94	121.32	121.71	122.09	122.48	122.86
60	123.24	123.63	124.01	124.39	124.78	125.16	125.54	125.93	126.31	126.69
70	127.07	127.46	127.84	128.22	128.60	128.99	129.37	129.75	130.13	130.51
80	130.89	131.28	131.66	132.04	132.42	132.80	133.18	133.56	133.94	134.32
90	134.70	135.08	135.46	135.84	136.22	136.60	136.98	137.36	137.74	138.12
100	138.50	138.88	139.26	139.64	140.02	140.40	140.77	141.15	141.53	141.91
110	142.29	142.67	143.04	143.42	143.80	144.18	144.55	144.93	145.31	145.69
120	146.06	146.44	146.82	147.19	147.57	147.95	148.32	148.70	149.07	149.45
130	149.83	150.20	150.58	150.95	151.33	151.70	152.08	152.45	152.83	153.20
140	153.58	153.95	154.33	154.70	155.08	155.45	155.83	156.20	156.57	156.95
150	157.32	157.69	158.07	158.44	158.81	159.19	159.56	159.93	160.30	160.68
160	161.05	161.42	161.79	162.16	162.53	162.91	163.28	163.65	164.02	164.39
170	164.76	165.13	165.50	165.88	166.25	166.62	166.99	167.36	167.73	168.10
180	168.47	168.84	169.21	169.58	169.95	170.31	170.68	171.05	171.42	171.79
190	172.16	172.53	172.90	173.26	173.63	174.00	174.37	174.74	175.10	175.47
200	175.84	176.21	176.58	176.94	177.31	177.68	178.04	178.41	178.78	179.14
210	179.51	179.88	180.24	180.61	180.98	181.34	181.71	182.07	182.44	182.81
220	183.17	183.54	183.90	184.27	184.63	185.00	185.36	185.73	186.09	186.45
230	186.82	187.18	187.55	187.91	188.27	188.64	189.00	189.37	189.73	190.09
240	190.46	190.82	191.18	191.54	191.91	192.27	192.63	192.99	193.36	193.72
250	194.08	194.44	194.80	195.17	195.53	195.89	196.25	196.61	196.97	197.33
260	197.69	198.05	198.41	198.77	199.14	199.50	199.86	200.22	200.58	200.94
270	201.29	201.65	202.01	202.37	202.73	203.09	203.45	203.81	204.17	204.53
280	204.88	205.24	205.60	205.96	206.32	206.68	207.03	207.39	207.75	208.11
290	208.46	208.82	209.18	209.53	209.89	210.25	210.60	210.96	211.32	211.67
°C	0	1	2	3	4	5	6	7	8	9

°C	0	1	2	3	4	5	6	7	8	9
300	212.03	212.39	212.74	213.10	213.45	213.81	214.16	214.52	214.88	215.23
310	215.59	215.94	216.29	216.65	217.00	217.36	217.71	218.07	218.42	218.77
320	219.13	219.48	219.84	220.19	220.54	220.90	221.25	221.60	221.96	222.31
330	222.66	223.01	223.37	223.72	224.07	224.42	224.77	225.13	225.48	225.83
340	226.18	226.53	226.88	227.24	227.59	227.94	228.29	228.64	228.99	229.34
350	229.69	230.04	230.39	230.74	231.09	231.44	231.79	232.14	232.49	232.84
360	233.19	233.54	233.89	234.23	234.58	234.93	235.28	235.63	235.98	236.32
370	236.67	237.02	237.37	237.72	238.06	238.41	238.76	239.11	239.45	239.80
380	240.15	240.49	240.84	241.19	241.53	241.88	242.23	242.57	242.92	243.26
390	243.61	243.95	244.30	244.65	244.99	245.34	245.68	246.03	246.37	246.72
400	247.06	247.40	247.75	248.09	248.44	248.78	249.12	249.47	249.81	250.16
410	250.50	250.84	251.19	251.53	251.87	252.21	252.56	252.90	253.24	253.58
420	253.93	254.27	254.61	254.95	255.29	255.64	255.98	256.32	256.66	257.00
430	257.34	257.68	258.02	258.37	258.71	259.05	259.39	259.73	260.07	260.41
440	260.75	261.09	261.43	261.77	262.11	262.45	262.78	263.12	263.46	263.80
450	264.14	264.48	264.82	265.16	265.49	265.83	266.17	266.51	266.85	267.18
460	267.52	267.86	268.20	268.53	268.87	269.21	269.54	269.88	270.22	270.55
470	270.89	271.23	271.56	271.90	272.24	272.57	272.91	273.24	273.58	273.91
480	274.25	274.58	274.92	275.25	275.59	275.92	276.26	276.59	276.93	277.26
490	277.60	277.93	278.26	278.60	278.93	279.26	279.60	279.93	280.26	280.60
500	280.93	281.26	281.60	281.93	282.26	282.59	282.93	283.26	283.59	283.92
510	284.25	284.58	284.92	285.25	285.58	285.91	286.24	286.57	286.90	287.23
520	287.56	287.90	288.23	288.56	288.89	289.22	289.55	289.88	290.21	290.54
530	290.86	291.19	291.52	291.85	292.18	292.51	292.84	293.17	293.50	293.82
540	294.15	294.48	294.81	295.14	295.47	295.79	296.12	296.45	296.78	297.10
550	297.43	297.76	298.08	298.41	298.74	299.07	299.39	299.72	300.04	300.37
560	300.70	301.02	301.35	301.68	302.00	302.33	302.65	302.98	303.30	303.63
570	303.95	304.28	304.60	304.93	305.25	305.58	305.90	306.22	306.55	306.87
580	307.20	307.52	307.84	308.17	308.49	308.81	309.14	309.46	309.78	310.11
590	310.43	310.75	311.07	311.40	311.72	312.04	312.36	312.68	313.01	313.33
600	313.65	313.97	314.29	314.61	314.93	315.26	315.58	315.90	316.22	316.54
610	316.86	317.18	317.50	317.82	318.14	318.46	318.78	319.10	319.41	319.73
620	320.05	320.37	320.69	321.01	321.33	321.65	321.96	322.28	322.60	322.92
630	323.24	323.55	323.87	324.19	324.51	324.82	325.14	325.46	325.78	326.09
640	326.41	326.73	327.04	327.36	327.67	327.99	328.31	328.62	328.94	329.25
650	329.57	329.89	330.20	330.52	330.83	331.15	331.46	331.78	332.09	332.41
660	332.72	333.04	333.35	333.66	333.98	334.29	334.61	334.92	335.23	335.55
670	335.86	336.17	336.49	336.80	337.11	337.43	337.74	338.05	338.36	338.68
680	338.99	339.30	339.61	339.92	340.24	340.55	340.86	341.17	341.48	341.79
690	342.11	342.42	342.73	343.04	343.35	343.66	343.97	344.28	344.59	344.90
700	345.21	345.52	345.83	346.14	346.45	346.76	347.07	347.38	347.68	347.99
710	348.30	348.61	348.92	349.23	349.53	349.84	350.15	350.46	350.77	351.07
720	351.38	351.69	352.00	352.30	352.61	352.92	353.22	353.53	353.84	354.14
730	354.45	354.75	355.06	355.37	355.67	355.98	356.28	356.59	356.89	357.20
740	357.51	357.81	358.12	358.42	358.72	359.03	359.33	359.64	359.94	360.25
750	360.55	360.85	361.16	361.46	361.77	362.07	362.37	362.68	362.98	363.28
760	363.59	363.89	364.19	364.49	364.80	365.10	365.40	365.70	366.01	366.31
770	366.61	366.91	367.21	367.51	367.81	368.12	368.42	368.72	369.02	369.32
780	369.62	369.92	370.22	370.52	370.82	371.12	371.42	371.72	372.02	372.32
790	372.62	372.92	373.22	373.52	373.82	374.12	374.42	374.71	375.01	375.31
800	375.61	375.91	376.21	376.50	376.80	377.10	377.40	377.70	377.99	378.29
810	378.59	378.88	379.18	379.48	379.77	380.07	380.37	380.66	380.96	381.26
820	381.55	381.85	382.14	382.44	382.74	383.03	383.33	383.62	383.92	384.21
830	384.51	384.80	385.10	385.39	385.69	385.98	386.27	386.57	386.86	387.16
840	387.45	387.74	388.04	388.33	388.62	388.92	389.21	389.50	389.79	390.09

°C 0 1 2 3 4 5 6 7 8 9

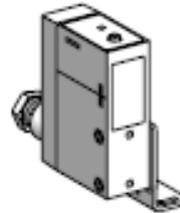
Anexo 2

References,
characteristics

Photo-electric sensors

Osiris® Application, materials handling series
With analogue output signal 4...20 mA and 0...10 V ⁽¹⁾
d.c. supply. Solid-state output

Compact design



System		Diffuse
Type of transmission		Infrared
Nominal sensing distance (Sn)		20...80 cm
References		
3-wire	PNP	XUJ K803530
Weight (kg)		0.100
Characteristics		
Product certifications		CE, CSA, UL
Ambient air temperature	For operation	- 25...+ 60 °C
	For storage	- 40...+ 80 °C
Vibration resistance	Conforming to IEC 60068-2-6	7 gn, amplitude ± 1.5 mm (f = 10...55 Hz)
Shock resistance	Conforming to IEC 60068-2-27	20 gn, duration 11 ms
Degree of protection	Conforming to IEC 60529	IP 67
	Conforming to NF C 20-010	IP 671
Connection		Screw terminals, maximum capacity: 2 x 1.5 mm ² or 1 x 2.5 mm ²
Materials		Case: PEI ⁽²⁾
Rated supply voltage		— 24 V with protection against reverse polarity
Voltage limits (including ripple)		— 20...30 V
Output current	Maximum	20 mA
	Minimum	4 mA
Output voltage (Vs)		— 0...10 V
Output voltage drift in relation to temperature		< 10% between - 25 and + 80 °C
Output voltage drift in relation to object colour		< 10%
Current consumption, no-load		< 35 mA
Maximum switching frequency		10 Hz (for an output voltage variation of 1 V)
Delays	First-up	< 150 ms
Indicator light		The brightness of the LED is proportional to the output voltage

⁽¹⁾ Applications: position control, monitoring concentricity or eccentricity, closed loop regulation, monitoring displacement, etc.

⁽²⁾ PEI: high quality synthetic resin providing excellent withstand to mechanical shocks, vibration and the effects of external agents frequently encountered in industry: alcohol, acids, petroleum, oils, greases, washing agents (diluted sodium carbonate 4%, nitric acid 2%), formaldehyde vapour, splashing hydrochloric acid, etc.

Accessories on
page 3701202

30150-EN_Ver7.0.fm/2



Curves,
dimensions,
schemes,
connections

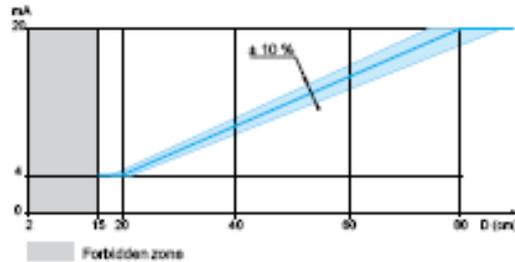
Photo-electric sensors

Osiris® Application, materials handling series
With analogue output signal 4...20 mA and 0...10 V (1)
d.c. supply. Solid-state output

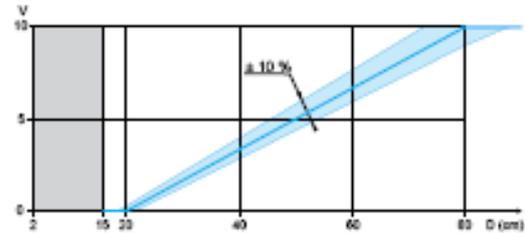
Curves

Output signal (related to distance of object). Test performed with 20 x 20 cm, white 80% object

Output current

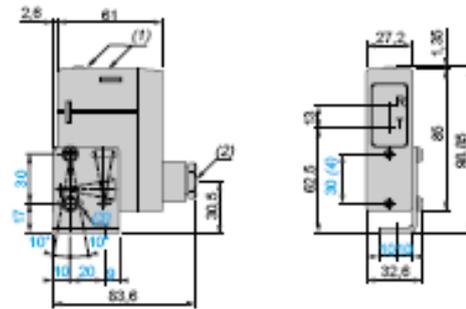


Output voltage



Dimensions

XUJ K803638

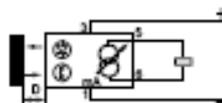


- (1) LED.
- (2) 11P cable gland.
- (3) \varnothing elongated hole \varnothing 4.2 x 14.
- (4) Front fixing (\varnothing 4 screws and inserts included).

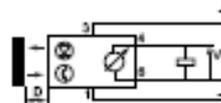
Wiring schemes

Diffuse system

Output current



Output voltage



Load characteristics

- Output current: the output current varies between 4 and 20 mA depending on the distance of the object and therefore, the load must be less than 1 k Ω .
- Output voltage: since the minimum rated output current of the sensor is 10 mA, the load must always have a resistive value of more than 1 k Ω .

Terminal connections

- 1 \ominus (-)
- 2 \oplus (+)
- 3 \oplus (+)
- 4 \ominus Voltage output
- 5 \ominus Current output
- 6 \ominus (-)

Terminals 1 and 6 connected internally.

Accessories
page 370/372

Anexo 3

Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 130 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
 - 8K Bytes of In-System Self-Programmable Flash
Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
In-System Programming by On-chip Boot Program
True Read-While-Write Operation
 - 512 Bytes EEPROM
Endurance: 100,000 Write/Erase Cycles
 - 1K Byte Internal SRAM
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Three PWM Channels
 - 8-channel ADC in TQFP and QFN/MLF package
Eight Channels 10-bit Accuracy
 - 6-channel ADC in PDIP package
Eight Channels 10-bit Accuracy
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-lead PDIP, 32-lead TQFP, and 32-pad QFN/MLF
- Operating Voltages
 - 2.7 - 5.5V (ATmega8L)
 - 4.5 - 5.5V (ATmega8)
- Speed Grades
 - 0 - 8 MHz (ATmega8L)
 - 0 - 16 MHz (ATmega8)
- Power Consumption at 4 Mhz, 3V, 25°C
 - Active: 3.6 mA
 - Idle Mode: 1.0 mA
 - Power-down Mode: 0.5 µA



8-bit AVR®
with 8K Bytes
In-System
Programmable
Flash

ATmega8
ATmega8L

24890-AVR-10/06





Interrupts

This section describes the specifics of the interrupt handling performed by the ATmega8. For a general explanation of the AVR interrupt handling, refer to "Reset and Interrupt Handling" on page 14.

Interrupt Vectors in ATmega8

Table 18. Reset and Interrupt Vectors

Vector No.	Program Address ⁽²⁾	Source	Interrupt Definition
1	0x000 ⁽¹⁾	RESET	External Pin, Power-on Reset, Brown-out Reset, and Watchdog Reset
2	0x001	INT0	External Interrupt Request 0
3	0x002	INT1	External Interrupt Request 1
4	0x003	TIMER2 COMP	Timer/Counter2 Compare Match
5	0x004	TIMER2 OVF	Timer/Counter2 Overflow
6	0x005	TIMER1 CAPT	Timer/Counter1 Capture Event
7	0x006	TIMER1 COMPA	Timer/Counter1 Compare Match A
8	0x007	TIMER1 COMPB	Timer/Counter1 Compare Match B
9	0x008	TIMER1 OVF	Timer/Counter1 Overflow
10	0x009	TIMER0 OVF	Timer/Counter0 Overflow
11	0x00A	SPI, STC	Serial Transfer Complete
12	0x00B	USART, RXC	USART, Rx Complete
13	0x00C	USART, UDRE	USART Data Register Empty
14	0x00D	USART, TXC	USART, Tx Complete
15	0x00E	ADC	ADC Conversion Complete
16	0x00F	EE_RDY	EEPROM Ready
17	0x010	ANA_COMP	Analog Comparator
18	0x011	TWI	Two-wire Serial Interface
19	0x012	SPM_RDY	Store Program Memory Ready

Notes: 1. When the BOOTRST Fuse is programmed, the device will jump to the Boot Loader address at reset, see "Boot Loader Support – Read-While-Write Self-Programming" on page 209.
 2. When the IVSEL bit in GICR is set, Interrupt Vectors will be moved to the start of the boot Flash section. The address of each Interrupt Vector will then be the address in this table added to the start address of the boot Flash section.

Table 19 shows reset and Interrupt Vectors placement for the various combinations of BOOTRST and IVSEL settings. If the program never enables an interrupt source, the Interrupt Vectors are not used, and regular program code can be placed at these locations. This is also the case if the Reset Vector is in the Application section while the Interrupt Vectors are in the boot section or vice versa.



Special Function IO Register – SFIOR

Bit	7	6	5	4	3	2	1	0	\$SFIOR
Read/Write	R	R	R	R	ACME	PUD	P&R2	P&R10	
Initial Value	0	0	0	0	0	0	0	0	

• Bit 2 – PUD: Pull-up Disable

When this bit is written to one, the pull-ups in the I/O ports are disabled even if the DDxn and PORTxn Registers are configured to enable the pull-ups ({DDxn, PORTxn} = 0b01). See "Configuring the Pin" on page 52 for more details about this feature.

Alternate Functions of Port B The Port B pins with alternate functions are shown in Table 22.

Table 22. Port B Pins Alternate Functions

Port Pin	Alternate Functions
PB7	XTAL2 (Chip Clock Oscillator pin 2) TOSC2 (Timer Oscillator pin 2)
PB6	XTAL1 (Chip Clock Oscillator pin 1 or External clock input) TOSC1 (Timer Oscillator pin 1)
PB5	SCK (SPI Bus Master clock Input)
PB4	MISO (SPI Bus Master Input/Slave Output)
PB3	MOSI (SPI Bus Master Output/Slave Input) OC2 (Timer/Counter2 Output Compare Match Output)
PB2	SS (SPI Bus Master Slave select) OC1B (Timer/Counter1 Output Compare Match B Output)
PB1	OC1A (Timer/Counter1 Output Compare Match A Output)
PB0	ICP1 (Timer/Counter1 Input Capture Pin)

The alternate pin configuration is as follows:

• XTAL2/TOSC2 – Port B, Bit 7

XTAL2: Chip clock Oscillator pin 2. Used as clock pin for crystal Oscillator or Low-frequency crystal Oscillator. When used as a clock pin, the pin can not be used as an I/O pin.

TOSC2: Timer Oscillator pin 2. Used only if internal calibrated RC Oscillator is selected as chip clock source, and the asynchronous timer is enabled by the correct setting in ASSR. When the AS2 bit in ASSR is set (one) to enable asynchronous clocking of Timer/Counter2, pin PB7 is disconnected from the port, and becomes the inverting output of the Oscillator amplifier. In this mode, a crystal Oscillator is connected to this pin, and the pin cannot be used as an I/O pin.

If PB7 is used as a clock pin, DDB7, PORTB7 and PINB7 will all read 0.

• XTAL1/TOSC1 – Port B, Bit 6

XTAL1: Chip clock Oscillator pin 1. Used for all chip clock sources except internal calibrated RC Oscillator. When used as a clock pin, the pin can not be used as an I/O pin.

TOSC1: Timer Oscillator pin 1. Used only if internal calibrated RC Oscillator is selected as chip clock source, and the asynchronous timer is enabled by the correct setting in ASSR. When the AS2 bit in ASSR is set (one) to enable asynchronous clocking of Timer/Counter2, pin PB6 is disconnected from the port, and becomes the input of the

ATmega8(L)

inverting Oscillator amplifier. In this mode, a crystal Oscillator is connected to this pin, and the pin can not be used as an I/O pin.

If PB6 is used as a clock pin, DDB6, PORTB6 and PINB6 will all read 0.

- **SCK – Port B, Bit 5**

SCK: Master Clock output, Slave Clock input pin for SPI channel. When the SPI is enabled as a Slave, this pin is configured as an input regardless of the setting of DDB5. When the SPI is enabled as a Master, the data direction of this pin is controlled by DDB5. When the pin is forced by the SPI to be an input, the pull-up can still be controlled by the PORTB5 bit.

- **MISO – Port B, Bit 4**

MISO: Master Data input, Slave Data output pin for SPI channel. When the SPI is enabled as a Master, this pin is configured as an input regardless of the setting of DDB4. When the SPI is enabled as a Slave, the data direction of this pin is controlled by DDB4. When the pin is forced by the SPI to be an input, the pull-up can still be controlled by the PORTB4 bit.

- **MOSI/OC2 – Port B, Bit 3**

MOSI: SPI Master Data output, Slave Data input for SPI channel. When the SPI is enabled as a Slave, this pin is configured as an input regardless of the setting of DDB3. When the SPI is enabled as a Master, the data direction of this pin is controlled by DDB3. When the pin is forced by the SPI to be an input, the pull-up can still be controlled by the PORTB3 bit.

OC2, Output Compare Match Output: The PB3 pin can serve as an external output for the Timer/Counter2 Compare Match. The PB3 pin has to be configured as an output (DDB3 set (one)) to serve this function. The OC2 pin is also the output pin for the PWM mode timer function.

- **\overline{SS} /OC1B – Port B, Bit 2**

\overline{SS} : Slave Select input. When the SPI is enabled as a Slave, this pin is configured as an input regardless of the setting of DDB2. As a Slave, the SPI is activated when this pin is driven low. When the SPI is enabled as a Master, the data direction of this pin is controlled by DDB2. When the pin is forced by the SPI to be an input, the pull-up can still be controlled by the PORTB2 bit.

OC1B, Output Compare Match output: The PB2 pin can serve as an external output for the Timer/Counter1 Compare Match B. The PB2 pin has to be configured as an output (DDB2 set (one)) to serve this function. The OC1B pin is also the output pin for the PWM mode timer function.

- **OC1A – Port B, Bit 1**

OC1A, Output Compare Match output: The PB1 pin can serve as an external output for the Timer/Counter1 Compare Match A. The PB1 pin has to be configured as an output (DDB1 set (one)) to serve this function. The OC1A pin is also the output pin for the PWM mode timer function.

- **ICP1 – Port B, Bit 0**

ICP1 – Input Capture Pin: The PB0 pin can act as an Input Capture Pin for Timer/Counter1.

Table 23 and Table 24 relate the alternate functions of Port B to the overriding signals shown in Figure 25 on page 56. SPI MSTR INPUT and SPI SLAVE OUTPUT constitute the MISO signal, while MOSI is divided into SPI MSTR OUTPUT and SPI SLAVE INPUT.



Table 23. Overriding Signals for Alternate Functions in PB7..PB4

Signal Name	PB7/XTAL2/ TOSC2 ⁽¹⁾⁽²⁾	PB6/XTAL1/ TOSC1 ⁽¹⁾	PB5/SCK	PB4/MISO
PUOE	$\overline{\text{EXT}} \cdot (\overline{\text{INTRC}} + \text{AS2})$	$\overline{\text{INTRC}} + \text{AS2}$	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \overline{\text{MSTR}}$
PUO	0	0	$\text{PORTB5} \cdot \overline{\text{PUD}}$	$\text{PORTB4} \cdot \overline{\text{PUD}}$
DDOE	$\overline{\text{EXT}} \cdot (\overline{\text{INTRC}} + \text{AS2})$	$\overline{\text{INTRC}} + \text{AS2}$	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \overline{\text{MSTR}}$
DDOV	0	0	0	0
PVOE	0	0	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \overline{\text{MSTR}}$
PVOV	0	0	SCK OUTPUT	SPI SLAVE OUTPUT
DIEOE	$\overline{\text{EXT}} \cdot (\overline{\text{INTRC}} + \text{AS2})$	$\overline{\text{INTRC}} + \text{AS2}$	0	0
DIEOV	0	0	0	0
DI	–	–	SCK INPUT	SPI MSTR INPUT
AIO	Oscillator Output	Oscillator/Clock Input	–	–

Notes: 1. INTRC means that the internal RC Oscillator is selected (by the CKSEL Fuse).
2. EXT means that the external RC Oscillator or an external clock is selected (by the CKSEL Fuse).

Table 24. Overriding Signals for Alternate Functions in PB3..PB0

Signal Name	PB3/MOS/OC2	PB2/SS/OC1B	PB1/OC1A	PB0/ICP1
PUOE	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \overline{\text{MSTR}}$	0	0
PUO	$\text{PORTB3} \cdot \overline{\text{PUD}}$	$\text{PORTB2} \cdot \overline{\text{PUD}}$	0	0
DDOE	$\text{SPE} \cdot \overline{\text{MSTR}}$	$\text{SPE} \cdot \overline{\text{MSTR}}$	0	0
DDOV	0	0	0	0
PVOE	$\text{SPE} \cdot \overline{\text{MSTR}} + \text{OC2 ENABLE}$	OC1B ENABLE	OC1A ENABLE	0
PVOV	$\text{SPI MSTR OUTPUT} + \text{OC2}$	OC1B	OC1A	0
DIEOE	0	0	0	0
DIEOV	0	0	0	0
DI	SPI SLAVE INPUT	SPI SS	–	ICP1 INPUT
AIO	–	–	–	–

ATmega8(L)

USART

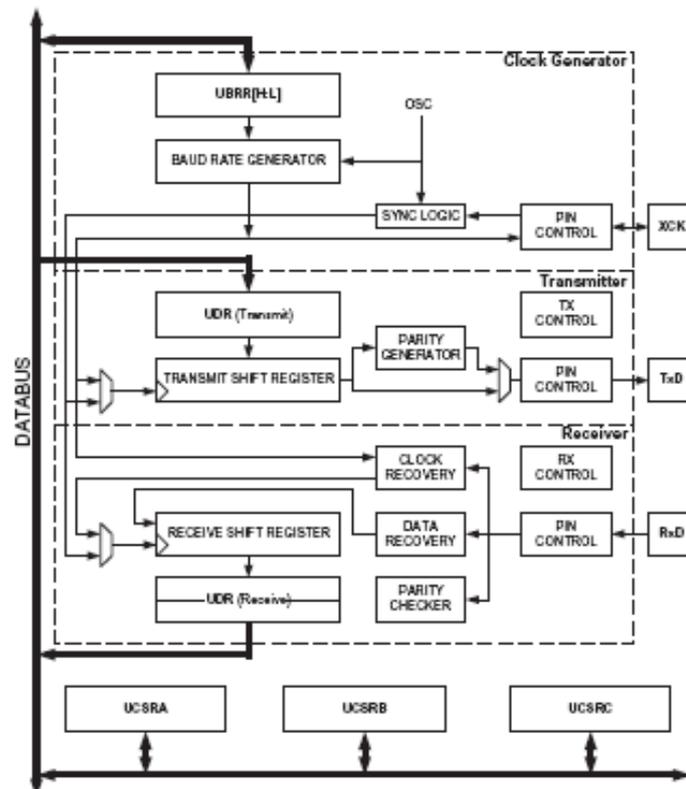
The Universal Synchronous and Asynchronous serial Receiver and Transmitter (USART) is a highly-flexible serial communication device. The main features are:

- Full Duplex Operation (Independent Serial Receive and Transmit Registers)
- Asynchronous or Synchronous Operation
- Master or Slave Clocked Synchronous Operation
- High Resolution Baud Rate Generator
- Supports Serial Frames with 5, 6, 7, 8, or 9 Databits and 1 or 2 Stop Bits
- Odd or Even Parity Generation and Parity Check Supported by Hardware
- Data OverRun Detection
- Framing Error Detection
- Noise Filtering Includes False Start Bit Detection and Digital Low Pass Filter
- Three Separate Interrupts on TX Complete, TX Data Register Empty and RX Complete
- Multi-processor Communication Mode
- Double Speed Asynchronous Communication Mode

Overview

A simplified block diagram of the USART Transmitter is shown in Figure 61. CPU accessible I/O Registers and I/O pins are shown in bold.

Figure 61. USART Block Diagram⁽¹⁾



Note: 1. Refer to "Pin Configurations" on page 2, Table 30 on page 64, and Table 29 on page 64 for USART pin placement.



The dashed boxes in the block diagram separate the three main parts of the USART (listed from the top): Clock generator, Transmitter and Receiver. Control Registers are shared by all units. The clock generation logic consists of synchronization logic for external clock input used by synchronous slave operation, and the baud rate generator. The XCK (transfer clock) pin is only used by synchronous transfer mode. The Transmitter consists of a single write buffer, a serial Shift Register, Parity Generator and control logic for handling different serial frame formats. The write buffer allows a continuous transfer of data without any delay between frames. The Receiver is the most complex part of the USART module due to its clock and data recovery units. The recovery units are used for asynchronous data reception. In addition to the recovery units, the Receiver includes a parity checker, control logic, a Shift Register and a two level receive buffer (UDR). The Receiver supports the same frame formats as the Transmitter, and can detect Frame Error, Data OverRun and Parity Errors.

AVR USART vs. AVR UART – Compatibility

The USART is fully compatible with the AVR UART regarding:

- Bit locations inside all USART Registers.
- Baud Rate Generation.
- Transmitter Operation.
- Transmit Buffer Functionality.
- Receiver Operation.

However, the receive buffering has two improvements that will affect the compatibility in some special cases:

- A second Buffer Register has been added. The two Buffer Registers operate as a circular FIFO buffer. Therefore the UDR must only be read once for each incoming data! More important is the fact that the Error Flags (FE and DOR) and the ninth data bit (RXB8) are buffered with the data in the receive buffer. Therefore the status bits must always be read before the UDR Register is read. Otherwise the error status will be lost since the buffer state is lost.
- The Receiver Shift Register can now act as a third buffer level. This is done by allowing the received data to remain in the serial Shift Register (see Figure 61) if the Buffer Registers are full, until a new start bit is detected. The USART is therefore more resistant to Data OverRun (DOR) error conditions.

The following control bits have changed name, but have same functionality and register location:

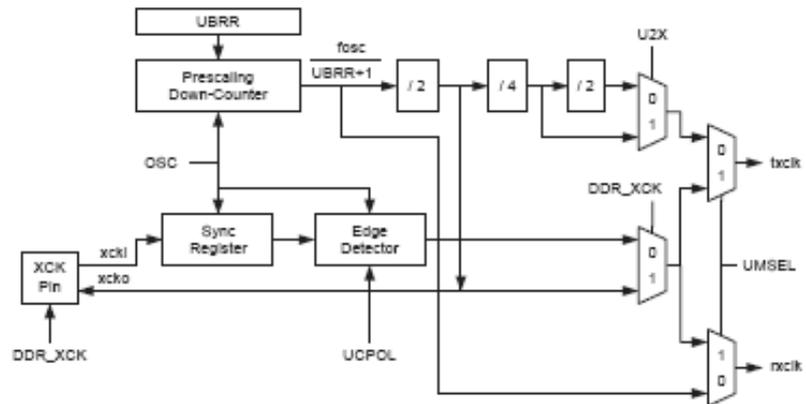
- CHR9 is changed to UCSZ2.
- OR is changed to DOR.

Clock Generation

The clock generation logic generates the base clock for the Transmitter and Receiver. The USART supports four modes of clock operation: normal asynchronous, double speed asynchronous, Master synchronous and Slave Synchronous mode. The UMSEL bit in USART Control and Status Register C (UCSRC) selects between asynchronous and synchronous operation. Double speed (Asynchronous mode only) is controlled by the U2X found in the UCSRA Register. When using Synchronous mode (UMSEL = 1), the Data Direction Register for the XCK pin (DDR_XCK) controls whether the clock source is internal (Master mode) or external (Slave mode). The XCK pin is only active when using Synchronous mode.

Figure 62 shows a block diagram of the clock generation logic.

Figure 62. Clock Generation Logic, Block Diagram



Signal description:

- txclk** Transmitter clock. (Internal Signal)
- rxclk** Receiver base clock. (Internal Signal)
- xcki** Input from XCK pin (internal Signal). Used for synchronous slave operation.
- xcko** Clock output to XCK pin (Internal Signal). Used for synchronous master operation.
- fosc** XTAL pin frequency (System Clock).

**Internal Clock Generation –
The Baud Rate Generator**

Internal clock generation is used for the asynchronous and the Synchronous Master modes of operation. The description in this section refers to Figure 62.

The USART Baud Rate Register (UBRR) and the down-counter connected to it function as a programmable prescaler or baud rate generator. The down-counter, running at system clock (fosc), is loaded with the UBRR value each time the counter has counted down to zero or when the UBRR Register is written. A clock is generated each time the counter reaches zero. This clock is the baud rate generator clock output (= fosc/(UBRR+1)). The Transmitter divides the baud rate generator clock output by 2, 8, or 16 depending on mode. The baud rate generator output is used directly by the Receiver's clock and data recovery units. However, the recovery units use a state machine that uses 2, 8, or 16 states depending on mode set by the state of the UMSEL, U2X and DDR_XCK bits.

Table 52 contains equations for calculating the baud rate (in bits per second) and for calculating the UBRR value for each mode of operation using an internally generated clock source.



Table 52. Equations for Calculating Baud Rate Register Setting

Operating Mode	Equation for Calculating Baud Rate ⁽¹⁾	Equation for Calculating UBRR Value
Asynchronous Normal mode (U2X = 0)	$BAUD = \frac{f_{OSC}}{16(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{16BAUD} - 1$
Asynchronous Double Speed Mode (U2X = 1)	$BAUD = \frac{f_{OSC}}{8(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{8BAUD} - 1$
Synchronous Master Mode	$BAUD = \frac{f_{OSC}}{2(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{2BAUD} - 1$

Note: 1. The baud rate is defined to be the transfer rate in bit per second (bps).

BAUD Baud rate (in bits per second, bps)

f_{OSC} System Oscillator clock frequency

UBRR Contents of the UBRRH and UBRRL Registers, (0 - 4095)

Some examples of UBRR values for some system clock frequencies are found in Table 60 (see page 159).

Double Speed Operation (U2X)

The transfer rate can be doubled by setting the U2X bit in UCSRA. Setting this bit only has effect for the asynchronous operation. Set this bit to zero when using synchronous operation.

Setting this bit will reduce the divisor of the baud rate divider from 16 to 8, effectively doubling the transfer rate for asynchronous communication. Note however that the Receiver will in this case only use half the number of samples (reduced from 16 to 8) for data sampling and clock recovery, and therefore a more accurate baud rate setting and system clock are required when this mode is used. For the Transmitter, there are no downsides.

External Clock

External clocking is used by the Synchronous Slave modes of operation. The description in this section refers to Figure 62 for details.

External clock input from the XCK pin is sampled by a synchronization register to minimize the chance of meta-stability. The output from the synchronization register must then pass through an edge detector before it can be used by the Transmitter and Receiver. This process introduces a two CPU clock period delay and therefore the maximum external XCK clock frequency is limited by the following equation:

$$f_{XCK} < \frac{f_{OSC}}{4}$$

Note that f_{OSC} depends on the stability of the system clock source. It is therefore recommended to add some margin to avoid possible loss of data due to frequency variations.

ATmega8(L)

Read Access

Doing a read access to the UBRRH or the UCSRC Register is a more complex operation. However, in most applications, it is rarely necessary to read any of these registers.

The read access is controlled by a timed sequence. Reading the I/O location once returns the UBRRH Register contents. If the register location was read in previous system clock cycle, reading the register in the current clock cycle will return the UCSRC contents. Note that the timed sequence for reading the UCSRC is an atomic operation. Interrupts must therefore be controlled (e.g., by disabling interrupts globally) during the read operation.

The following code example shows how to read the UCSRC Register contents.

Assembly Code Example ⁽¹⁾
<pre> USART_ReadUCSRC: ; Read UCSRC in r16,UBRRH in r16,UCSRC ret </pre>
C Code Example ⁽¹⁾
<pre> unsigned char USART_ReadUCSRC(void) { unsigned char ucsrc; /* Read UCSRC */ ucsrc = UBRRH; ucsrc = UCSRC; return ucsrc; } </pre>

Note: 1. See "About Code Examples" on page 8.

The assembly code example returns the UCSRC value in r16.

Reading the UBRRH contents is not an atomic operation and therefore it can be read as an ordinary register, as long as the previous instruction did not access the register location.

USART Register Description

USART I/O Data Register – UDR

Bit	7	6	5	4	3	2	1	0	
	RXB[7:0]								UDR (Read)
	TXB[7:0]								UDR (Write)
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

The USART Transmit Data Buffer Register and USART Receive Data Buffer Registers share the same I/O address referred to as USART Data Register or UDR. The Transmit Data Buffer Register (TXB) will be the destination for data written to the UDR Register location. Reading the UDR Register location will return the contents of the Receive Data Buffer Register (RXB).

For 5-, 6-, or 7-bit characters the upper unused bits will be ignored by the Transmitter and set to zero by the Receiver.



The transmit buffer can only be written when the UDRE Flag in the UCSRA Register is set. Data written to UDR when the UDRE Flag is not set, will be ignored by the USART Transmitter. When data is written to the transmit buffer, and the Transmitter is enabled, the Transmitter will load the data into the Transmit Shift Register when the Shift Register is empty. Then the data will be serially transmitted on the TxD pin.

The receive buffer consists of a two level FIFO. The FIFO will change its state whenever the receive buffer is accessed. Due to this behavior of the receive buffer, do not use Read-Modify-Write instructions (SBI and CBI) on this location. Be careful when using bit test instructions (SBIC and SBIS), since these also will change the state of the FIFO.

USART Control and Status Register A – UCSRA

Bit	7	6	5	4	3	2	1	0	
	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	UCSRA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

• **Bit 7 – RXC: USART Receive Complete**

This flag bit is set when there are unread data in the receive buffer and cleared when the receive buffer is empty (i.e. does not contain any unread data). If the Receiver is disabled, the receive buffer will be flushed and consequently the RXC bit will become zero. The RXC Flag can be used to generate a Receive Complete interrupt (see description of the RXCIE bit).

• **Bit 6 – TXC: USART Transmit Complete**

This flag bit is set when the entire frame in the Transmit Shift Register has been shifted out and there are no new data currently present in the transmit buffer (UDR). The TXC Flag bit is automatically cleared when a transmit complete interrupt is executed, or it can be cleared by writing a one to its bit location. The TXC Flag can generate a Transmit Complete interrupt (see description of the TXCIE bit).

• **Bit 5 – UDRE: USART Data Register Empty**

The UDRE Flag indicates if the transmit buffer (UDR) is ready to receive new data. If UDRE is one, the buffer is empty, and therefore ready to be written. The UDRE Flag can generate a Data Register Empty interrupt (see description of the UDRIE bit).

UDRE is set after a reset to indicate that the Transmitter is ready.

• **Bit 4 – FE: Frame Error**

This bit is set if the next character in the receive buffer had a Frame Error when received (i.e., when the first stop bit of the next character in the receive buffer is zero). This bit is valid until the receive buffer (UDR) is read. The FE bit is zero when the stop bit of received data is one. Always set this bit to zero when writing to UCSRA.

• **Bit 3 – DOR: Data OverRun**

This bit is set if a Data OverRun condition is detected. A Data OverRun occurs when the receive buffer is full (two characters), it is a new character waiting in the Receive Shift Register, and a new start bit is detected. This bit is valid until the receive buffer (UDR) is read. Always set this bit to zero when writing to UCSRA.

• **Bit 2 – PE: Parity Error**

This bit is set if the next character in the receive buffer had a Parity Error when received and the parity checking was enabled at that point (UPM1 = 1). This bit is valid until the receive buffer (UDR) is read. Always set this bit to zero when writing to UCSRA.

• **Bit 1 – U2X: Double the USART transmission speed**

ATmega8(L)

This bit only has effect for the asynchronous operation. Write this bit to zero when using synchronous operation.

Writing this bit to one will reduce the divisor of the baud rate divider from 16 to 8 effectively doubling the transfer rate for asynchronous communication.

- **Bit 0 – MPCM: Multi-processor Communication Mode**

This bit enables the Multi-processor Communication mode. When the MPCM bit is written to one, all the incoming frames received by the USART Receiver that do not contain address information will be ignored. The Transmitter is unaffected by the MPCM setting. For more detailed information see "Multi-processor Communication Mode" on page 151.

USART Control and Status Register B – UCSRB

Bit	7	6	5	4	3	2	1	0	
	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – RXCIE: RX Complete Interrupt Enable**

Writing this bit to one enables interrupt on the RXC Flag. A USART Receive Complete interrupt will be generated only if the RXCIE bit is written to one, the Global Interrupt Flag in SREG is written to one and the RXC bit in UCSRA is set.

- **Bit 6 – TXCIE: TX Complete Interrupt Enable**

Writing this bit to one enables interrupt on the TXC Flag. A USART Transmit Complete interrupt will be generated only if the TXCIE bit is written to one, the Global Interrupt Flag in SREG is written to one and the TXC bit in UCSRA is set.

- **Bit 5 – UDRIE: USART Data Register Empty Interrupt Enable**

Writing this bit to one enables interrupt on the UDRE Flag. A Data Register Empty interrupt will be generated only if the UDRIE bit is written to one, the Global Interrupt Flag in SREG is written to one and the UDRE bit in UCSRA is set.

- **Bit 4 – RXEN: Receiver Enable**

Writing this bit to one enables the USART Receiver. The Receiver will override normal port operation for the RxD pin when enabled. Disabling the Receiver will flush the receive buffer invalidating the FE, DOR and PE Flags.

- **Bit 3 – TXEN: Transmitter Enable**

Writing this bit to one enables the USART Transmitter. The Transmitter will override normal port operation for the TxD pin when enabled. The disabling of the Transmitter (writing TXEN to zero) will not become effective until ongoing and pending transmissions are completed (i.e., when the Transmit Shift Register and Transmit Buffer Register do not contain data to be transmitted). When disabled, the Transmitter will no longer override the TxD port.

- **Bit 2 – UCSZ2: Character Size**

The UCSZ2 bits combined with the UCSZ1:0 bit in UCSRC sets the number of data bits (Character Size) in a frame the Receiver and Transmitter use.

- **Bit 1 – RXB8: Receive Data Bit 8**

RXB8 is the ninth data bit of the received character when operating with serial frames with nine data bits. Must be read before reading the low bits from UDR.

- **Bit 0 – TXB8: Transmit Data Bit 8**



TXB8 is the ninth data bit in the character to be transmitted when operating with serial frames with nine data bits. Must be written before writing the low bits to UDR.

USART Control and Status Register C – UCSRC

Bit	7	6	5	4	3	2	1	0	
	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	0	0	0	0	1	1	0	

The UCSRC Register shares the same I/O location as the UBRRH Register. See the "Accessing UBRRH/UCSRC Registers" on page 152 section which describes how to access this register.

- **Bit 7 – URSEL: Register Select**

This bit selects between accessing the UCSRC or the UBRRH Register. It is read as one when reading UCSRC. The URSEL must be one when writing the UCSRC.

- **Bit 6 – UMSEL: USART Mode Select**

This bit selects between Asynchronous and Synchronous mode of operation.

Table 55. UMSEL Bit Settings

UMSEL	Mode
0	Asynchronous Operation
1	Synchronous Operation

ATmega8(L)

- **Bit 5:4 – UPM1:0: Parity Mode**

These bits enable and set type of Parity Generation and Check. If enabled, the Transmitter will automatically generate and send the parity of the transmitted data bits within each frame. The Receiver will generate a parity value for the incoming data and compare it to the UPM0 setting. If a mismatch is detected, the PE Flag in UCSRA will be set.

Table 56. UPM Bits Settings

UPM1	UPM0	Parity Mode
0	0	Disabled
0	1	Reserved
1	0	Enabled, Even Parity
1	1	Enabled, Odd Parity

- **Bit 3 – USBS: Stop Bit Select**

This bit selects the number of stop bits to be inserted by the transmitter. The Receiver ignores this setting.

Table 57. USBS Bit Settings

USBS	Stop Bit(s)
0	1-bit
1	2-bit

- **Bit 2:1 – UCSZ1:0: Character Size**

The UCSZ1:0 bits combined with the UCSZ2 bit in UCSRB sets the number of data bits (Character Size) in a frame the Receiver and Transmitter use.

Table 58. UCSZ Bits Settings

UCSZ2	UCSZ1	UCSZ0	Character Size
0	0	0	5-bit
0	0	1	6-bit
0	1	0	7-bit
0	1	1	8-bit
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Reserved
1	1	1	9-bit

- **Bit 0 – UCPOL: Clock Polarity**



This bit is used for Synchronous mode only. Write this bit to zero when Asynchronous mode is used. The UCPOL bit sets the relationship between data output change and data input sample, and the synchronous clock (XCK).

Table 59. UCPOL Bit Settings

UCPOL	Transmitted Data Changed (Output of TxD Pin)	Received Data Sampled (Input on RxD Pin)
0	Rising XCK Edge	Falling XCK Edge
1	Falling XCK Edge	Rising XCK Edge

USART Baud Rate Registers – UBRRL and UBRRHs

Bit	15	14	13	12	11	10	9	8	UBRRH
	URSEL	–	–	–	UBRR[11:8]				
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	

The UBRRH Register shares the same I/O location as the UCSRC Register. See the “Accessing UBRRH/UCSRC Registers” on page 152 section which describes how to access this register.

• **Bit 15 – URSEL: Register Select**

This bit selects between accessing the UBRRH or the UCSRC Register. It is read as zero when reading UBRRH. The URSEL must be zero when writing the UBRRH.

• **Bit 14:12 – Reserved Bits**

These bits are reserved for future use. For compatibility with future devices, these bit must be written to zero when UBRRH is written.

• **Bit 11:0 – UBRR11:0: USART Baud Rate Register**

This is a 12-bit register which contains the USART baud rate. The UBRRH contains the four most significant bits, and the UBRRL contains the eight least significant bits of the USART baud rate. Ongoing transmissions by the Transmitter and Receiver will be corrupted if the baud rate is changed. Writing UBRRL will trigger an immediate update of the baud rate prescaler.



Analog-to-Digital Converter

Features

- 10-bit Resolution
- 0.5 LSB Integral Non-linearity
- ± 2 LSB Absolute Accuracy
- 13 - 260 μ s Conversion Time
- Up to 15 kSPS at Maximum Resolution
- 6 Multiplexed Single Ended Input Channels
- 2 Additional Multiplexed Single Ended Input Channels (TQFP and QFNMLF Package only)
- Optional Left Adjustment for ADC Result Readout
- 0 - V_{CC} ADC Input Voltage Range
- Selectable 2.56V ADC Reference Voltage
- Free Running or Single Conversion Mode
- Interrupt on ADC Conversion Complete
- Sleep Mode Noise Canceler

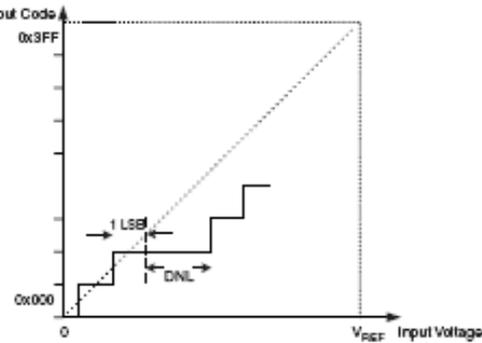
The ATmega8 features a 10-bit successive approximation ADC. The ADC is connected to an 8-channel Analog Multiplexer which allows eight single-ended voltage inputs constructed from the pins of Port C. The single-ended voltage inputs refer to 0V (GND).

The ADC contains a Sample and Hold circuit which ensures that the input voltage to the ADC is held at a constant level during conversion. A block diagram of the ADC is shown in Figure 90.

The ADC has a separate analog supply voltage pin, AV_{CC} . AV_{CC} must not differ more than $\pm 0.3V$ from V_{CC} . See the paragraph "ADC Noise Canceler" on page 201 on how to connect this pin.

Internal reference voltages of nominally 2.56V or AV_{CC} are provided On-chip. The voltage reference may be externally decoupled at the AREF pin by a capacitor for better noise performance.

Figure 100. Differential Non-linearity (DNL)



- **Quantization Error:** Due to the quantization of the input voltage into a finite number of codes, a range of input voltages (1 LSB wide) will code to the same value. Always ± 0.5 LSB.
- **Absolute accuracy:** The maximum deviation of an actual (unadjusted) transition compared to an ideal transition for any code. This is the compound effect of offset, gain error, differential error, non-linearity, and quantization error. Ideal value: ± 0.5 LSB.

ADC Conversion Result

After the conversion is complete (ADIF is high), the conversion result can be found in the ADC Result Registers (ADCL, ADCH).

For single ended conversion, the result is

$$ADC = \frac{V_{IN} - 1024}{V_{REF}}$$

where V_{IN} is the voltage on the selected input pin and V_{REF} the selected voltage reference (see Table 74 on page 206 and Table 75 on page 206). 0x000 represents ground, and 0x3FF represents the selected reference voltage minus one LSB.

ADC Multiplexer Selection Register – ADMUX

Bit	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	–	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

• **Bit 7:6 – REFS1:0: Reference Selection Bits**

These bits select the voltage reference for the ADC, as shown in Table 74. If these bits are changed during a conversion, the change will not go in effect until this conversion is complete (ADIF in ADCSRA is set). The internal voltage reference options may not be used if an external reference voltage is being applied to the AREF pin.



Table 74. Voltage Reference Selections for ADC

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal V_{REF} turned off
0	1	AV_{CC} with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56V Voltage Reference with external capacitor at AREF pin

• **BIT 5 – ADLAR: ADC Left Adjust Result**

The ADLAR bit affects the presentation of the ADC conversion result in the ADC Data Register. Write one to ADLAR to left adjust the result. Otherwise, the result is right adjusted. Changing the ADLAR bit will affect the ADC Data Register immediately, regardless of any ongoing conversions. For a complete description of this bit, see "The ADC Data Register – ADCL and ADCH" on page 208.

• **Bits 3:0 – MUX3:0: Analog Channel Selection Bits**

The value of these bits selects which analog inputs are connected to the ADC. See Table 75 for details. If these bits are changed during a conversion, the change will not go in effect until this conversion is complete (ADIF in ADCSRA is set).

Table 75. Input Channel Selections

MUX3..0	Single Ended Input
0000	ADC0
0001	ADC1
0010	ADC2
0011	ADC3
0100	ADC4
0101	ADC5
0110	ADC6
0111	ADC7
1000	
1001	
1010	
1011	
1100	
1101	
1110	1.23V (V_{DD})
1111	0V (GND)

ATmega8(L)

ADC Control and Status Register A – ADCSRA

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADFR	ADIF	ADIE	ADP82	ADP81	ADP80	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – ADEN: ADC Enable**

Writing this bit to one enables the ADC. By writing it to zero, the ADC is turned off. Turning the ADC off while a conversion is in progress, will terminate this conversion.

- **Bit 6 – ADSC: ADC Start Conversion**

In Single Conversion mode, write this bit to one to start each conversion. In Free Running mode, write this bit to one to start the first conversion. The first conversion after ADSC has been written after the ADC has been enabled, or if ADSC is written at the same time as the ADC is enabled, will take 25 ADC clock cycles instead of the normal 13. This first conversion performs initialization of the ADC.

ADSC will read as one as long as a conversion is in progress. When the conversion is complete, it returns to zero. Writing zero to this bit has no effect.

- **Bit 5 – ADFR: ADC Free Running Select**

When this bit is set (one) the ADC operates in Free Running mode. In this mode, the ADC samples and updates the Data Registers continuously. Clearing this bit (zero) will terminate Free Running mode.

- **Bit 4 – ADIF: ADC Interrupt Flag**

This bit is set when an ADC conversion completes and the Data Registers are updated. The ADC Conversion Complete Interrupt is executed if the ADIE bit and the I-bit in SREG are set. ADIF is cleared by hardware when executing the corresponding interrupt Handling Vector. Alternatively, ADIF is cleared by writing a logical one to the flag. Beware that if doing a Read-Modify-Write on ADCSRA, a pending interrupt can be disabled. This also applies if the SBI and CBI instructions are used.

- **Bit 3 – ADIE: ADC Interrupt Enable**

When this bit is written to one and the I-bit in SREG is set, the ADC Conversion Complete Interrupt is activated.



• Bits 2:0 – ADPS2:0: ADC Prescaler Select Bits

These bits determine the division factor between the XTAL frequency and the input clock to the ADC.

Table 76. ADC Prescaler Selections

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

The ADC Data Register – ADCL and ADCH

ADLAR = 0

Bit	15	14	13	12	11	10	9	8	
	–	–	–	–	–	–	ADC8	ADC8	ADCH
	ADC7	ADC8	ADC6	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

ADLAR = 1

Bit	15	14	13	12	11	10	9	8	
	ADC8	ADC8	ADC7	ADC8	ADC6	ADC4	ADC3	ADC2	ADCH
	ADC1	ADC0	–	–	–	–	–	–	ADCL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

When an ADC conversion is complete, the result is found in these two registers.

When ADCL is read, the ADC Data Register is not updated until ADCH is read. Consequently, if the result is left adjusted and no more than 8-bit precision is required, it is sufficient to read ADCH. Otherwise, ADCL must be read first, then ADCH.

The ADLAR bit in ADMUX, and the MUXn bits in ADMUX affect the way the result is read from the registers. If ADLAR is set, the result is left adjusted. If ADLAR is cleared (default), the result is right adjusted.

• ADC9:0: ADC Conversion result

These bits represent the result from the conversion, as detailed in "ADC Conversion Result" on page 205.

Anexo 4

```

; *****
;
; Programa que sensa los valores de los ADC del microcontrolador y envía
; dichos valores serialmente mediante una trama
; *****
;

.include "D:\ARCHIV~1\VMLAB\include\m8def.inc"

; Define here the variables

.def TEMP          = R16
.def TEMP1         = R17
.def BCDUnidad     = R18
.def BCDDecena     = R19
.def BCDCentena   = R20
.def ContInter     = R21
.def contCaracter  = R22
.def FIN           = R23
.def IGUALES       = R24

.equ CR = 13          ;Caracter ENTER
.equ SI = 1
.equ NO = 0
.equ PORTCONTROL = PORTB
.equ DDRCONTROL  = DDRB

.DSEG
.org $60

ValorADC0      : .byte 1
ValorADC1      : .byte 1
ValorADC2      : .byte 1
ValorADC3      : .byte 1
ValorADC4      : .byte 1
ValorADC5      : .byte 1
ValorADCBBCD   : .byte 1
CadenaRx       : .byte 14

.CSEG

reset:
    RJMP      start
    reti      ; Addr $01
    reti      ; Addr $02
    reti      ; Addr $03
    reti      ; Addr $04
    reti      ; Addr $05
    RJMP      Inter100ms
    reti      ; Addr $07
  
```

```

reti                ; Addr $08
reti                ; Addr $09
reti                ; Addr $0A
RJMP                RxCaracter
reti                ; Addr $0C
reti                ; Addr $0D
reti                ; Addr $0E
reti                ; Addr $0F
reti                ; Addr $10

```

start:

```

LDI                TEMP,LOW(RAMEND)
OUT                SPL,TEMP
LDI                TEMP,HIGH(RAMEND)
OUT                SPH,TEMP

```

```

LDI                TEMP,0b00111111
OUT                DDRCONTROL,TEMP
LDI                contCaracter,0

```

```

RCALL              ConfiguraSerial
RCALL              ConfiguraInter100ms

```

```

LDI                YL,low(CadenaRx)
LDI                YH,high(CadenaRx)
SEI

```

;**PROGRAMA PRINCIPAL**

Inicio:

```

RCALL              RecibeCaracter
CPI                TEMP,'i'
BRNE              Inicio
LDI                YL,low(CadenaRx)
LDI                YH,high(CadenaRx)
LDI                contCaracter,0

```

SigueLeyendoADC:

```

RCALL              LeeADC0
RCALL              LeeADC1
RCALL              LeeADC2
RCALL              LeeADC3
RCALL              LeeADC4
RCALL              LeeADC5

```

```

LDI                ZL,low(ProtocoloInicioSensor1*2)
LDI                ZH,high(ProtocoloInicioSensor1*2)
RCALL              EnviaCadena

```

```

LDS      TEMP,ValorADC0
RCALL    EnviaValorADC

LDI      ZL,low(ProtocoloInicioSensor2*2)
LDI      ZH,high(ProtocoloInicioSensor2*2)
RCALL    EnviaCadena
LDS      TEMP,ValorADC1
RCALL    EnviaValorADC

LDI      ZL,low(ProtocoloInicioSensor3*2)
LDI      ZH,high(ProtocoloInicioSensor3*2)
RCALL    EnviaCadena
LDS      TEMP,ValorADC2
RCALL    EnviaValorADC

LDI      ZL,low(ProtocoloInicioSensor4*2)
LDI      ZH,high(ProtocoloInicioSensor4*2)
RCALL    EnviaCadena
LDS      TEMP,ValorADC3
RCALL    EnviaValorADC

LDI      ZL,low(ProtocoloInicioSensor5*2)
LDI      ZH,high(ProtocoloInicioSensor5*2)
RCALL    EnviaCadena
LDS      TEMP,ValorADC4
RCALL    EnviaValorADC

LDI      ZL,low(ProtocoloInicioSensor6*2)
LDI      ZH,high(ProtocoloInicioSensor6*2)
RCALL    EnviaCadena
LDS      TEMP,ValorADC5
RCALL    EnviaValorADC

LDI      ContInter,0
LDI      YL,low(CadenaRx)
LDI      YH,high(CadenaRx)
LDI      contCaracter,0
SigueEsperando:
LDS      TEMP,CadenaRx
CPI      TEMP,'f'
BREQ     Inicio
CPI      ContInter,5      ;espera que pase 2 seg
BRNE     SigueEsperando
RJMP     SigueLeyendoADC

;***FIN PROGRAMA PRINCIPAL***

```

```

;*****
;
; Subrutina de interrupción de recepción de carácter
;*****

```

```

RxCaracter:
    INC        contCaracter
    IN         TEMP,UDR
    ST         Y+,TEMP
    CPI        contCaracter,13
    BRNE      FinRxCaracter
    RCALL     VerificaFin
    CPI        FIN,SI
    BRNE      FinRxCaracter
    RCALL     VerificaProtocoloControl

```

```

FinRxCaracter:
    RETI

```

```

;*****
;
; Subrutina que verifica que en protocolo de control termine en END
;
;
; Salida: FIN
;*****

```

```

VerificaFin:
    LDI        FIN,NO
    SUBI       YL,3
    LD         TEMP,Y+
    CPI        TEMP,'E'
    BRNE      FinVerificaFin
    LD         TEMP,Y+
    CPI        TEMP,'N'
    BRNE      FinVerificaFin
    LD         TEMP,Y+
    CPI        TEMP,'D'
    BRNE      FinVerificaFin
    LDI        FIN,SI
    LDI        contCaracter,0

```

```

FinVerificaFin:
    RET

```

```

;*****
;
; Verifica que el protocolo de control enviado sea el correcto
; (CONTROLxONEND o CONTROLxOFEND)
;*****

```

```

VerificaProtocoloControl:
    LDI        ZL,low(ProtocoloControl1ON*2)
    LDI        ZH,high(ProtocoloControl1ON*2)
    RCALL     ComparaCadenas
    CPI        IGUALES,SI
    BRNE      ComparaControl1OFF
    SBI       PORTCONTROL,0
    RJMP     FinProtocoloControl

```

ComparaControl1OFF:

LDI	ZL,low(ProtocoloControl1OFF*2)
LDI	ZH,high(ProtocoloControl1OFF*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl2ON
CBI	PORTCONTROL,0
RJMP	FinProtocoloControl

ComparaControl2ON:

LDI	ZL,low(ProtocoloControl2ON*2)
LDI	ZH,high(ProtocoloControl2ON*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl2OFF
SBI	PORTCONTROL,1
RJMP	FinProtocoloControl

ComparaControl2OFF:

LDI	ZL,low(ProtocoloControl2OFF*2)
LDI	ZH,high(ProtocoloControl2OFF*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl3ON
CBI	PORTCONTROL,1
RJMP	FinProtocoloControl

ComparaControl3ON:

LDI	ZL,low(ProtocoloControl3ON*2)
LDI	ZH,high(ProtocoloControl3ON*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl3OFF
SBI	PORTCONTROL,2
RJMP	FinProtocoloControl

ComparaControl3OFF:

LDI	ZL,low(ProtocoloControl3OFF*2)
LDI	ZH,high(ProtocoloControl3OFF*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl4ON
CBI	PORTCONTROL,2

ComparaControl4ON:

LDI	ZL,low(ProtocoloControl4ON*2)
LDI	ZH,high(ProtocoloControl4ON*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI

BRNE	ComparaControl4OFF
SBI	PORTCONTROL,3
RJMP	FinProtocoloControl

ComparaControl4OFF:

LDI	ZL,low(ProtocoloControl4OFF*2)
LDI	ZH,high(ProtocoloControl4OFF*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl5ON
CBI	PORTCONTROL,3
RJMP	FinProtocoloControl

ComparaControl5ON:

LDI	ZL,low(ProtocoloControl5ON*2)
LDI	ZH,high(ProtocoloControl5ON*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl5OFF
SBI	PORTCONTROL,4
RJMP	FinProtocoloControl

ComparaControl5OFF:

LDI	ZL,low(ProtocoloControl5OFF*2)
LDI	ZH,high(ProtocoloControl5OFF*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl6ON
CBI	PORTCONTROL,4
RJMP	FinProtocoloControl

ComparaControl6ON:

LDI	ZL,low(ProtocoloControl6ON*2)
LDI	ZH,high(ProtocoloControl6ON*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	ComparaControl6OFF
SBI	PORTCONTROL,5
RJMP	FinProtocoloControl

ComparaControl6OFF:

LDI	ZL,low(ProtocoloControl6OFF*2)
LDI	ZH,high(ProtocoloControl6OFF*2)
RCALL	ComparaCadenas
CPI	IGUALES,SI
BRNE	FinProtocoloControl
CBI	PORTCONTROL,5

FinProtocoloControl:

LDI	YL,low(CadenaRx)
-----	------------------

```

LDI      YH,high(CadenaRx)
LDI      contCaracter,0
RET

;*****
;
;Compara dos cadenas (Tabla y cadena en DSEG)
;
;Entrada:   Z (apunta al inicio de la cadena en Tabla)
;           Y (apunta al inicio de la cadena en DSEG)
;Salida: COMPARACION
;Valores Modificados: TEMP,TEMP1,Z,Y,COMPARACION,contCaracteres
;*****
ComparaCadenas:
LDI      YL,low(CadenaRx)
LDI      YH,high(CadenaRx)
LDI      IGUALES,NO
SigueComparaCadenas:
LPM      TEMP,Z+
CPI      TEMP,0
BREQ     SonIguales
LD       TEMP1,Y+
CP       TEMP,TEMP1
BREQ     SigueComparaCadenas
RJMP     FinComparaCadenas
SonIguales:
LDI      IGUALES,SI
FinComparaCadenas:
RET

;*****
;
;Configura la interrupción por tiempo a 100ms
;
;Valores modificados: TEMP
;*****
ConfigurarInter100ms:
LDI      TEMP,0b00000000
OUT      TCCR1A,TEMP
LDI      TEMP,0b00001100 ;100 PARA 100ms
OUT      TCCR1B,TEMP
LDI      TEMP,LOW(3125) ;3125 PARA 100 ms
LDI      R17,HIGH(3125)
OUT      OCR1AH,R17
OUT      OCR1AL,TEMP
LDI      TEMP,$10
OUT      TIMSK,TEMP
RET

```

```

*****
;
; Subrutina que lleva la cuenta de 100ms
;
;
; Salida: ContInter
*****
Inter100ms:
    INC        ContInter
    RETI
;
*****
;
; Envia el valor del ADCx almacenado en TEMP por serial
; junto con el fin del protocolo (END)
;
;
; Entrada: TEMP (valor del ADCx)
; Salida: ValorADCBCD
; Valores modificados: Z
*****
EnviaValorADC:
    RCALL     ConvierteNumeroaBCD
    LDS      TEMP,ValorADCBCD
    RCALL     EnviaCaracter
    LDS      TEMP,ValorADCBCD+1
    RCALL     EnviaCaracter
    LDS      TEMP,ValorADCBCD+2
    RCALL     EnviaCaracter
    LDI      ZL,low(ProtocoloFin*2)
    LDI      ZH,high(ProtocoloFin*2)
    RCALL     EnviaCadena
    RET
;
*****
;
; Configura el ADC0
;
;
; Valores modificados: TEMP
*****
ConfiguraADC0:
    LDI      TEMP,0b01100000
    OUT     ADMUX,TEMP
    LDI      TEMP,0b11000110
    OUT     ADCSR,TEMP
    RET
;
*****
;
; Configura el ADC1
;
;
; Valores modificados: TEMP
*****
ConfiguraADC1:
    LDI      TEMP,0b011000001
    OUT     ADMUX,TEMP

```

```

      LDI      TEMP,0b11000110
      OUT     ADCSR,TEMP
      RET

;*****
;
; Configura el ADC2
;
;
; Valores modificados: TEMP
;*****
ConfiguraADC2:
      LDI      TEMP,0b01100010
      OUT     ADMUX,TEMP
      LDI      TEMP,0b11000110
      OUT     ADCSR,TEMP
      RET

;*****
;
; Configura el ADC3
;
;
; Valores modificados: TEMP
;*****
ConfiguraADC3:
      LDI      TEMP,0b01100011
      OUT     ADMUX,TEMP
      LDI      TEMP,0b11000110
      OUT     ADCSR,TEMP
      RET

;*****
;
; Configura el ADC4
;
;
; Valores modificados: TEMP
;*****
ConfiguraADC4:
      LDI      TEMP,0b01100100
      OUT     ADMUX,TEMP
      LDI      TEMP,0b11000110
      OUT     ADCSR,TEMP
      RET

;*****
;
; Configura el ADC5
;
;
; Valores modificados: TEMP
;*****
ConfiguraADC5:
      LDI      TEMP,0b01100101
      OUT     ADMUX,TEMP
      LDI      TEMP,0b11000110
      OUT     ADCSR,TEMP
  
```

RET

```

*****
;
; Lee el valor del ADC0 y lo almacena en ValorADC0
;
;
; Valores modificados: ValorADC0
*****
LeeADC0:
    RCALL    ConfiguraADC0
SigueLeeADC0:
    SBIS    ADCSR,ADIF
    RJMP    SigueLeeADC0
    IN      TEMP,ADCH
    STS     ValorADC0,TEMP
    CBI     ADCSR,7
    RET

*****
;
; Lee el valor del ADC1 y lo almacena en ValorADC1
;
;
; Valores modificados: ValorADC1
*****
LeeADC1:
    RCALL    ConfiguraADC1
SigueLeeADC1:
    SBIS    ADCSR,ADIF
    RJMP    SigueLeeADC1
    IN      TEMP,ADCH
    STS     ValorADC1,TEMP
    CBI     ADCSR,7
    RET

*****
;
; Lee el valor del ADC2 y lo almacena en ValorADC2
;
;
; Valores modificados: ValorADC2
*****
LeeADC2:
    RCALL    ConfiguraADC2
SigueLeeADC2:
    SBIS    ADCSR,ADIF
    RJMP    SigueLeeADC2
    IN      TEMP,ADCH
    STS     ValorADC2,TEMP
    CBI     ADCSR,7
    RET

```

```

*****
;
; Lee el valor del ADC3 y lo almacena en ValorADC3
;
;
; Valores modificados: ValorADC3
*****
;
LeeADC3:
    RCALL    ConfiguraADC3
SigueLeeADC3:
    SBIS     ADCSR,ADIF
    RJMP     SigueLeeADC3
    IN       TEMP,ADCH
    STS     ValorADC3,TEMP
    CBI     ADCSR,7
    RET

```

```

*****
;
; Lee el valor del ADC4 y lo almacena en ValorADC4
;
;
; Valores modificados: ValorADC4
*****
;
LeeADC4:
    RCALL    ConfiguraADC4
SigueLeeADC4:
    SBIS     ADCSR,ADIF
    RJMP     SigueLeeADC4
    IN       TEMP,ADCH
    STS     ValorADC4,TEMP
    CBI     ADCSR,7
    RET

```

```

*****
;
; Lee el valor del ADC5 y lo almacena en ValorADC5
;
;
; Valores modificados: ValorADC5
*****
;
LeeADC5:
    RCALL    ConfiguraADC5
SigueLeeADC5:
    SBIS     ADCSR,ADIF
    RJMP     SigueLeeADC5
    IN       TEMP,ADCH
    STS     ValorADC5,TEMP
    CBI     ADCSR,7
    RET

```

```

*****
;
; Configura la comunicación serial
;
; Frecuencia      : 8 Mhz
; Velocidad      : 9600 bps
; Bits de datos   : 8
; Paridad        : ninguno
; Bits de parada  : 1
*****
ConfiguraSerial:
    LDI    TEMP,0<<U2X
    OUT   UCSRA,TEMP
    LDI    TEMP,high(51)
    OUT   UBRRH,TEMP
    LDI    TEMP,low(51)
    OUT   UBRRL,TEMP
    LDI    TEMP,0b10011000 ;RXEN=1 y TXEN=1
    OUT   UCSRB,TEMP
    LDI    TEMP,0b10000110
    OUT   UCSRC,TEMP
    RET

*****
;
; Envía una cadena por comunicación serial
;
; Entrada: Puntero Z (inicio de la cadena)
; Valores modificados: TEMP
*****
EnviaCadena:
    LPM    TEMP,Z+
    CPI    TEMP,0
    BREQ   FinEnviaCaracter
    RCALL  EnviaCaracter
    RJMP   EnviaCadena
FinEnviaCaracter:
    RET

*****
;
; Envía un caracter por comunicación serial
;
; Entrada: TEMP
*****
EnviaCaracter:
    SBIS   UCSRA,UDRE
    RJMP   EnviaCaracter
    OUT   UDR,TEMP
    RET

```

```

*****
;
; Recibe un caracter por comunicación serial
;
;
; Salida: TEMP
*****
RecibeCaracter:
    SBIS        UCSRA,RXC
    RJMP        RecibeCaracter
    IN          TEMP,UDR
    RET

*****
;
; Recibe un caracter por comunicación serial
; sin esperar que haya alguno
;
;
; Salida: TEMP
*****
RecibeCaracterSinEspera:
    SBIS        UCSRA,RXC
    RJMP        FinRecibeCaracter
    IN          TEMP,UDR
FinRecibeCaracter:
    RET

*****
;
; Convierte un número de un registro a BCD
; TEMP:BCDL    TEMP1:BCDH
;
;
; Entrada:    SMS
; Salida:     RespuestaPetición
*****
ConvierteNumeroaBCD:
    PUSH        BCDUnidad
    PUSH        BCDDecena
    PUSH        BCDCentena
    PUSH        TEMP
    PUSH        TEMP1

    MOV         BCDUnidad,TEMP
    CLR         BCDDecena
    CLR         BCDCentena
SigueConvierteBCD:
    SUBI        BCDUnidad,10
    BRCS        FinConvierteBCD
    CPI         BCDDecena,9
    BRSH        IncrementaCentenas
    INC         BCDDecena
    RJMP        SigueConvierteBCD
IncrementaCentenas:
    INC         BCDCentena
  
```

```

LDI      BCDDecena,0
RJMP     SigueConvierteBCD
FinConvierteBCD:
SUBI     BCDUnidad,-10

RCALL    BorraValorADC
LDI      TEMP1,48
ADD      BCDCentena,TEMP1
ADD      BCDDecena,TEMP1
ADD      BCDUnidad,TEMP1
STS      ValorADCBCD,BCDCentena
STS      ValorADCBCD+1,BCDDecena
STS      ValorADCBCD+2,BCDUnidad

POP      TEMP1
POP      TEMP
POP      BCDCentena
POP      BCDDecena
POP      BCDUnidad

RET

;*****
;
; Inicializa con espacios en blanco la cadena ValorADC
;
; Valores Modificados: Y
;*****
BorraValorADC:
PUSH     TEMP
PUSH     TEMP1

LDI      TEMP,0
LDI      yl,low(ValorADCBCD)
LDI      yh,high(ValorADCBCD)

SigueBorraValorADC:
CPI      TEMP,3
BREQ     FinBorraValorADC
INC      TEMP
LDI      TEMP1,''
ST       Y+,TEMP1
RJMP     SigueBorraValorADC

FinBorraValorADC:

POP      TEMP1
POP      TEMP
RET

```

.,***TABLA***

; Protocolo con el que envia los datos VALORx:XXXEND

ProtocoloInicioSensor1:

.DB "VALOR1:",0

ProtocoloInicioSensor2:

.DB "VALOR2:",0

ProtocoloInicioSensor3:

.DB "VALOR3:",0

ProtocoloInicioSensor4:

.DB "VALOR4:",0

ProtocoloInicioSensor5:

.DB "VALOR5:",0

ProtocoloInicioSensor6:

.DB "VALOR6:",0

ProtocoloFin:

.DB "END",0

; Protocolo que recibe para el control

ProtocoloControl1ON:

.DB "CONTROL1ONEND",0

ProtocoloControl1OFF:

.DB "CONTROL1OFEND",0

ProtocoloControl2ON:

.DB "CONTROL2ONEND",0

ProtocoloControl2OFF:

.DB "CONTROL2OFEND",0

ProtocoloControl3ON:

.DB "CONTROL3ONEND",0

ProtocoloControl3OFF:

.DB "CONTROL3OFEND",0

ProtocoloControl4ON:

.DB "CONTROL4ONEND",0

ProtocoloControl4OFF:

.DB "CONTROL4OFEND",0

ProtocoloControl5ON:
.DB "CONTROL5ONEND",0

ProtocoloControl5OFF:
.DB "CONTROL5OFEND",0

ProtocoloControl6ON:
.DB "CONTROL6ONEND",0

ProtocoloControl6OFF:
.DB "CONTROL6OFEND",0

