



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA ELECTRÓNICA

**“ESTUDIO DE SISTEMAS WEB – EMBEBIDOS APLICADOS AL
DESARROLLO DE UN PROTOTIPO QUE CONTROLE
TEMPERATURA Y HUMEDAD”**

TESIS DE GRADO

**PREVIA LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y COMPUTACIÓN**

**PRESENTADO POR:
BRAVO MOROCHO FRANKLIN PATRICIO
LÓPEZ VILLA MARÍA VERÓNICA**

RIOBAMBA – ECUADOR

2009

A Dios, por haberme bendecido con una familia que constituye el pilar que me sostiene en el camino. A mis maestros, por su empeño en formar profesionales con valores que permitan un mejor futuro. A mis amigos y amigas porque me han ofrecido alegría y sobre todo su amistad incondicional. A todos ellos, GRACIAS.....

Verónica

*El presente trabajo lo dedico a todos los estudiantes
que están en constante superación. No desistan.*

Verónica

NOMBRE	FIRMA	FECHA
Dr. Ms.c. Romeo Rodríguez DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
Ing. Paúl Romero DIRECTOR DE LA ESCUELA DE INGENIERÍA ELECTRÓNICA Y TECNOLOGÍA EN COMPUTACIÓN.
Ing. Hugo Moreno DIRECTOR DE TESIS
Ing. Alberto Arrellano MIEMBRO DEL TRIBUNAL
Tlgo. Carlos Rodríguez DIRECTOR DPTO. DOCUMENTACION
NOTA DE LA TESIS

“Nosotros, **MARÍA VERÓNICA LÓPEZ VILLA**, y **FRANKLIN PATRICIO BRAVO MOROCHO**, somos responsables de las ideas, doctrinas y resultados expuestos en esta: tesis, y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo”

María Verónica López Villa

Franklin Patricio Bravo Morocho

ÍNDICE DE ABREVIATURAS

ADC	Convertidor Analógico Digital
ARP	Protocolo de Resolución de Direcciones
DMA	Acceso directo a memoria
ICMP	Protocolo de Mensajes de Control en Internet
IP	Protocolo de Internet
I2C	Circuitos Inter-Integrados
LAN	Red de Área Local
LLC	Control de Enlace Lógico
MAC	Control de Acceso al Medio
MCU	Unidad de Microcontrolador
MTU	Unidad Máxima de Transmisión
NTC	Coficiente Negativo de temperatura
OSI	Sistemas de Interconexión Abierta
PDU	Unidad de Datos de Protocolo
PHY	Transceptor de capa Física
PTC	Coficiente Positivo de Temperatura
RFC	Solicitud de comentario
SE	Sistemas Embebidos
SS	Selección de esclavo
SPI	Bus Serial de Interfaz de Periféricos
TCP	Protocolo de Control de Transporte
UDP	Protocolo de Datagrama de Usuario

ÍNDICE GENERAL

CAPÍTULO I: GENERALIDADES

1.1 Antecedentes.....	15
1.2 Justificación del Proyecto	16
1.3 Objetivos.....	17

CAPITULO II: FUNDAMENTO TEÓRICO

2.1. Redes de Datos	18
2.2. LAN, WAN E Internetworks	19
2.3. Modelo OSI	19
2.4. Descripción de Capas.....	20
2.5. Modelo TCP/IP	21
2.6. Protocolos más comunes.....	22
2.6.1. ARP, Address Resolution Protocol	22
2.6.2. IP, Internet Protocol.	23
2.6.3. ICMP, Internet Control Message Protocol.....	25
2.6.3.1. Comando Ping	26
2.6.4. TCP, Transfer Control Protocol	27
2.6.4.1. Encabezado TCP	27
2.6.4.2. Establecimiento de la conexión TCP.....	29
2.6.5. HTTP, Hypertext Transfer Protocol	31
2.6.5.1. Transacción HTTP.....	31
2.6.5.2. Solicitud HTTP	31
2.6.5.3. Respuesta HTTP	32
2.6.5.4. Métodos HTTP	32
2.7. Microcontroladores.....	33
2.7.1. Arquitecturas de Proceso	33
2.7.2. La Unidad Central de Proceso O CPU.....	34
2.7.3. Memorias.....	35

2.7.4. Puertos de entrada y salida.....	35
2.7.5. Recursos especiales del microcontrolador.....	36
2.8. SPI (serial peripheral interface).....	37
2.8.1. Especificaciones del bus.....	37
2.8.2. Ventajas	38
2.8.3. Desventajas	38
2.9. Modulo lcd	39
2.9.1. Tipos de displays LCD	39
2.10. Sensores	40
2.10.1. Tipos de Sensores.....	40
2.10.2. Sensor de temperatura	41
2.10.2.1. Temperatura	41
2.10.2.2. Tipos de Sensores de Temperatura	42
2.10.2.3. Aplicaciones de los Sensores de Temperatura	43
2.10.3. Sensor de humedad.....	43
2.10.3.1. Humedad Relativa	43
2.10.3.2. Tipos de Sensores de Temperatura	43
2.10.3.3. Efectos de la temperatura y la humedad.....	44
2.10.3.4. Aplicaciones de los Sensores de Humedad.....	44
2.11. Actuadores.....	45
2.11.1. Ventilación	46
2.11.2. Calefacción	47

CAPITULO III: ESTUDIO DE LOS SISTEMAS WEB EMBEBIDOS

3.1. Sistemas Embebidos.....	48
3.1.1. Partes de un SE	48
3.1.2. Ventajas al utilizar SE.....	49
3.1.3. Desventajas de los SE	50
3.1.4. Aplicaciones	50
3.2. Sistemas Web Embebidos	51
3.3. Selección del sistema web embebido.....	52
3.3.1. Alternativa ethernet	52

3.3.1.1. Wiznet W5100	52
3.3.1.2. PACKET WHACKER	53
3.3.1.3. Controlador ENC28J60	54
3.3.2. Métricas para la elección del módulo ethernet	55
3.3.3. Estudio de la mejor alternativa	56
3.3.4. Microcontrolador	58
3.3.4.1. Microcontroladores Microchip 16F87X	58
3.3.4.2. Microcontrolador ATMEL ATmega32	59
3.3.5. Métricas para la elección del microcontrolador.	61
3.3.6. Estudio de la mejor alternativa de microprocesador	61
3.3.7. Resultados de la elección	64

CAPITULO IV:

DISEÑO Y DESARROLLO DEL SISTEMA WEB EMBEBIDO

3.4. Software de desarrollo	64
3.4.1. AVR STUDIO	64
3.4.2. PROGISP	67
4.1. Requerimientos del sistema	69
4.2. Diseño e implementación de hardware	71
4.2.1. Adquisición de datos	72
4.2.1.1 Sensor de humedad	72
4.2.1.2. Sensor de temperatura	74
4.2.2. Microcontrolador	76
4.2.3. Control del sistema	77
4.3.2.1. Control dirección IP	78
4.3.2.2. Visualización	79
4.3.2.3. Interfaz de potencia	80
4.3.3. Diseño de la interfaz con la red	81
4.3. Diseño e implementación de software	82
4.3.1. Programa principal del sistema	83
4.3.2. Proceso para el controlador ethernet	85
4.3.4. Proceso servidor	88

4.3.3.1. Implementación del protocolo ARP.....	89
4.3.3.2. Implementación del protocolo IP.....	90
4.3.3.3. Implementación del protocolo ICMP.....	91
4.3.3.4. Implementación del protocolo TCP.....	92
4.3.3.5. Implementación del protocolo HTTP	93
4.3.5. Proceso de visualización	95
4.3.6. Proceso de control.....	95

CAPITULO V:

ANÁLISIS Y RESULTADOS

5.1. Medición de temperatura y humedad	97
5.2. Funcionamiento de los protocolos implementados	99
5.2.1. Protocolo ARP	100
5.2.2. Protocolo ICMP	100
5.2.3. Protocolo TCP	101
5.2.4. Protocolo HTTP.....	102

CONCLUSIONES

RECOMENDACIONES

RESUMEN

SUMMARY

GLOSARIO

ANEXOS

BIBLIOGRAFÍA

ÍNDICE DE FIGURAS

Figura II.1. Capas de Modelo OSI	20
Figura II.2. Capas del modelo TCP / IP	22
Figura II.3. Formato de paquete ARP de solicitud o reenvió con Ethernet.	22
Figura II.4 . Cabecera de un datagrama IP	24
Figura II.5. Mensaje ICMP Echo Request y Echo Reply	26
Figura II.6. Campos del encabezado TCP	28
Figura II.7. Proceso conexión TCP	30
Figura II.8. Diálogo HTTP	31
Figura II.9. Asignación de líneas Maestro – Esclavo	37
Figura II.10. Tipos de LCD	39
Figura II.11. Actuador Eléctrico: Servomotor	45
Figura II.12. Ventilador de 12V y 12Cm	46
Figura II.13. Caloventilador eléctrico	47
Figura III.14. Modelo de un Sistema Embebido	48
Figura III.15. Modulo Wiznet W5100	53
Figura III.16. Packet Whacker	53
Figura III.17. Controlador ENC28J60	54
Figura III.18. ENC28J60. Encapsulado SPDIP, SSOP, SOIC	55
Figura III.19. Pines del PIC16F87X	59
Figura III.20. AVR ATmega32 y su disposición de pines	60
Figura III.21. Pantalla de elección de tipo de proyecto	64
Figura III.22. Pantalla principal del AVR Studio	65
Figura III.23. Área de información de proyecto ejecutado	66
Figura III.24. Opciones de configuración del proyecto	67
Figura III.25. Software Programador PROGISP	67
Figura IV.26. Sistema Web Embebido implementado	69
Figura IV.27. Diagrama de bloques del Servidor Web Embebido	71
Figura IV.28. Multivibrador astable usando el HS1101	72
Figura IV.29. Salida del multivibrador astable	72
Figura IV.30. Circuito puente utilizado como interfaz al microprocesador	74
Figura IV.31. Diagrama de bloques interno del atmega32	76
Figura IV.32. Diagrama de hardware para el microprocesador	77
Figura IV.33. Control para la configuración de la IP	78

Figura IV.34. Conexiones físicas del LCD	81
Figura IV.35. Interfaz de Potencia a través de un Octoacoplador	81
Figura IV.36. Conexión entre el ATMEGA32 y ENC28J60	82
Figura IV.37. Diagrama de conexiones del ENC28J60	83
Figura IV.38. Diagrama de flujo principal del sistema	85
Figura IV.39. Esquema general de la función Server_process().....	89
Figura IV.40. Diagrama de Flujo del protocolo HTTP.....	94
Figura V.41. DAP-II (DATA AIRE INC)	99
Figura V.42. Mediciones de temperatura.....	99
Figura V.43. Captura del trafico ARP (Reply)	101
Figura V.44. Conexión verificada mediante Ping.....	102
Figura V.45. Flujo TCP.....	102
Figura V.46. Pagina Web visualizada en el navegador.....	103

ÍNDICE DE TABLAS

Tabla II.1. Métodos en la solicitud http.....	34
Tabla II.2. Sensores de Humedad y Temperatura	43
Tabla II.3. Sensores de temperatura en el mercado	43
Tabla II.4. Sensores De Humedad en el mercado.....	45
Tabla III.5. Asignación cuantitativa para las métricas.....	58
Tabla III.6. Precios de dispositivos Ethernet.....	58
Tabla III.7. Calificaciones y costos	59
Tabla III.8. Matriz de puntos.....	59
Tabla III.9. Asignación cuantitativa para las métricas.....	63
Tabla IV.10. Precios de los microcontroladores	64
Tabla III.11. Calificaciones y costos	64
Tabla III.12. Matriz de puntos	65
Tabla IV.13. Punto de Referencia a 6660Hz para 55% RH @ 25°C	75
Tabla IV.14. Valores de la frecuencia cuando RH = 0% hasta RH = 99%	75
Tabla IV.15. Valores ADC desde Tt = 0 hasta Tt = 99	77
Tabla IV.16. Descripción de pines del LCD 2X16	81
Tabla IV.17. Configuraciones de los Puertos.....	88
Tabla IV.18. Valores necesarios para generar el mensaje ARP	92
Tabla IV.19. Valores utilizados en la cabecera IP	92
Tabla IV.20. Parámetros para una petición ICMP Echo.....	93
Tabla IV.21. Valores asignados para el saludo de 3 vías	95
Tabla V.22. Mediciones DAP-II vs Sistema Web Embebido	101

INTRODUCCIÓN

Las redes de computadores hoy en día tienen gran importancia porque permiten la comunicación entre dispositivos de diferentes partes del mundo. Este beneficio se puede aprovechar para controlar, monitorear remotamente. Desde la comodidad de una oficina se puede conocer el estado de ciertos equipos ubicados a larga distancia.

Otra de las nuevas tecnologías que está logrando mayor avance es el mundo de los sistemas embebidos. Estos sistemas están destinados a realizar una o un grupo de tareas dedicadas a un fin específico. La principal ventaja de utilizar este tipo de sistemas es su bajo costo.

El presente proyecto consta de dos partes fundamentales: la primera de un estudio de los sistemas Web embebidos y la segunda parte se relaciona con la aplicación de esta tecnología en un prototipo que controle temperatura y humedad.

El estudio se enfocó en determinar las ventajas y desventajas que tiene el implementar circuitos con este tipo de sistemas, además de determinar el sistema adecuado para el prototipo que se construyó.

Los valores medidos de temperatura y humedad pueden ser accedidos desde una página Web almacenada en el sistema Web embebido digitando la dirección IP asignado al dispositivo y desde el cual se pudo controlar dos equipos orientados a mantener la temperatura y humedad según un rango.

Este proyecto precisó aplicar muchos de los conceptos estudiados a lo largo de la carrera de ingeniería electrónica, así como el estudio y aplicación de otros no abarcados por ella.

CAPITULO I

GENERALIDADES

1.1 Antecedentes

Un Sistema Embebido es aquel que integra en su circuitería un sistema digital reprogramable o reconfigurable. Estos sistemas son considerados como un todo porque combinan periféricos de entrada, un procesador interno, software de ejecución y periféricos de salida, además poseen una cantidad de recursos físicos, como memorias, disco duro, etc.

Sin duda que con el advenimiento de esta nueva tecnología, su alta escala de integración, riqueza de los recursos que integran sus circuitos integrados y el abaratamiento de los costos ha permitido ver cada vez más soluciones embebidas en medios electrónicos, de automatización y control de procesos.

El querer tener al alcance de la mano la información ha sido siempre un problema y se ha convertido casi en una necesidad. Muchas veces esta información se

encuentra a grandes distancias y no siempre se cuenta con el tiempo para esperar a que sea transportada hasta donde se requiere.

Ethernet es la tecnología de red LAN más usada, resultando idóneas para aquellos casos en los que se necesita una red local que deba transportar tráfico esporádico y ocasionalmente pesado a velocidades muy elevadas.

Gracias al avance tecnológico, hoy en día se puede controlar ciertos parámetros del clima como temperatura y humedad.

Es necesario también elegir una alternativa tecnológica de bajo costo que de paso al crecimiento económico

1.2 Justificación del Proyecto

El control y monitorización de equipos industriales y comerciales de manera remota se hace cada vez más necesario, incluso se podría decir que imprescindible. El contar con un sistema completamente dedicado a tales funciones se ha vuelto demasiado costoso.

La idea de tener una PC entera para este fin no es factible, terminando en una subutilización de los recursos, pues solo era posible utilizar una PC como puente entre el sistema de control y la red de datos.

Esta comunicación a distancia se ha facilitado con el desarrollo de una serie de dispositivos denominados sistemas embebidos los mismos que cumplen con niveles de automatización y eficiencia económica, los cuales ponen en primera línea la importancia de estudiar y utilizar estos dispositivos en este proyecto, pretendiendo que sea el pilar fundamental para futuros desarrollos en el mundo de los circuitos embebidos.

Además este tipo de sistemas se pueden utilizar en otras áreas y aplicaciones, dentro de la institución se esta desarrollando un proyecto utilizando sistemas embebidos para el laboratorio de comunicaciones.

1.3 Objetivos

1.3.1 Objetivo General

- Realizar un estudio de los sistemas Web embebidos aplicados en el desarrollo de un prototipo que controle temperatura y humedad.

1.3.2 Objetivos Específicos

- Estudiar los sistemas embebidos entendiendo su importancia, límites, componentes y requerimientos de diseño.
- Definir métricas que permitan la selección del sistema embebido más adecuado.
- Diseñar e implementar el circuito electrónico para la adquisición de datos.
- Diseñar e implementar el sistema de control.
- Desarrollar el software para el sistema.
- Analizar los resultados.

CAPITULO II

FUNDAMENTO TEÓRICO

2.1. Redes de Datos

Las Redes en la actualidad se han convertido en una parte esencial en la vida personal y comercial al permitir la comunicación en forma confiable con todos en todas partes. Estas redes de información varían en tamaño y capacidad, pero todas las redes tienen cuatro elementos básicos en común:

- **Protocolos.**- Reglas y acuerdos para regular cómo se envían, redireccionan, reciben e interpretan los mensajes.
- **Mensajes.**- Unidades de información que viajan de un dispositivo a otro.
- **Medio de comunicación.**- Una forma de interconectar esos dispositivos, que puede transportar los mensajes de un dispositivo a otro.
- **Dispositivos.**- los dispositivos de la red que cambian mensajes entre sí.

Para las redes de datos, la comunicación deberá estar destinada a ser exitosa. Sin embargo, debido a que un mensaje se traslada por la red, muchos factores pueden evitar que el mensaje llegue al receptor o distorsionar el significado pretendido como: la calidad de la ruta entre el emisor y el receptor, tamaño del mensaje, etc.

2.2. LAN, WAN e Internetworks

LAN o Red de área Local.- Por lo general se encuentran administradas por una organización única y cubren una determinada área geográfica proporcionando servicios y aplicaciones a personas dentro de una organización común.

Wan o Red de área amplia.- Cuando una organización tiene ubicaciones separadas por grandes distancias geográficas, se requerirá un proveedor de servicio de telecomunicaciones para interconectar las LAN. Este tipo de redes utilizan dispositivos de red diseñados específicamente para realizar las interconexiones para poder compartir recursos.

Internetworks.- La internetwork más conocida, ampliamente utilizada y a la que accede el público en general es Internet. Internet se crea por la interconexión de redes que pertenecen a los Proveedores de servicios de Internet (ISP). Estas redes ISP se conectan entre sí para proporcionar acceso a millones de usuarios en todo el mundo.

2.3. Modelo OSI

El modelo de interconexión de sistema abierto (OSI) es el modelo de referencia de internetwork más ampliamente conocido. Se utiliza para el diseño de redes de datos, especificaciones de funcionamiento y resolución de problemas.

Aunque los modelos TCP/IP y OSI son los modelos principales que se utilizan cuando se analiza la funcionalidad de red, los diseñadores de protocolos de red, servicios o dispositivos pueden crear sus propios modelos para representar sus productos. Por último, se solicita a los diseñadores que se comuniquen con la

industria asociando sus productos o servicios con el modelo OSI, el modelo TCP/IP o ambos.

OSI define para su modelo siete capas (Figura II.1.) cada una de ella agrega funciones que deben tener lugar para que las comunicaciones sean exitosas.

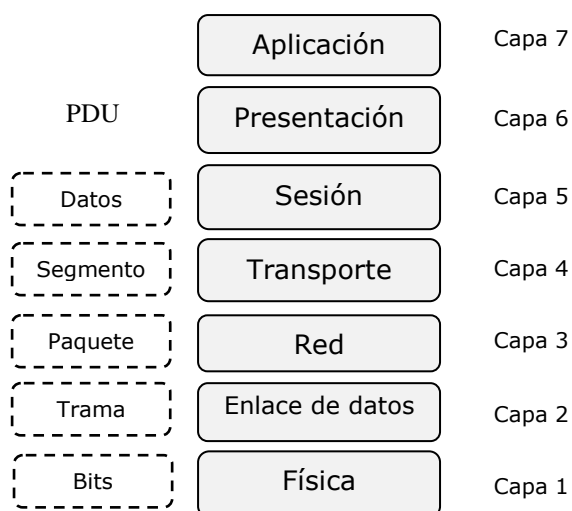


Figura II.1. Capas de Modelo OSI

2.4. Descripción de capas

- **Capa 7:** Esta es la capa que interactúa con el sistema operativo o aplicación cuando el usuario decide transferir archivos, leer mensajes, o realizar otras actividades de red. Por ello, en esta capa se incluyen tecnologías tales como http, DNS, SMTP, SSH, Telnet, etc.
- **Capa 6:** Esta capa tiene la misión de recoger los datos que han sido entregados por la capa de aplicación, y convertirlos en un formato estándar que otras capas puedan entender. En esta capa tenemos como ejemplo los formatos MP3, MPG, GIF, etc.

- **Capa 5:** Esta capa establece, mantiene y termina las comunicaciones que se forman entre dispositivos. Se pueden poner como ejemplo las sesiones SQL, RPC, NetBIOS, etc.
- **Capa 4:** Esta capa mantiene el control de flujo de datos, y provee de verificación de errores y recuperación de datos entre dispositivos. Control de flujo significa que la capa de transporte vigila si los datos vienen de más de una aplicación e integra cada uno de los datos de aplicación en un solo flujo dentro de la red física. Como ejemplos más claros tenemos TCP y UDP.
- **Capa 3:** Esta capa determina la forma en que serán mandados los datos al dispositivo receptor. Aquí se manejan los protocolos de enrutamiento y el manejo de direcciones IP. En esta capa hablamos de IP, IPX, X.25, etc.
- **Capa 2:** También llamada capa de enlaces de datos. En esta capa, el protocolo físico adecuado es asignado a los datos. Se asigna el tipo de red y la secuencia de paquetes utilizada. Los ejemplos más claros son Ethernet, ATM, Frame Relay, etc.
- **Capa 1:** Este es el nivel de lo que llamamos hardware. Define las características físicas de la red, como las conexiones, niveles de voltaje, cableado, etc. Se pueden incluir en esta capa la fibra óptica, el par trenzado, cable cruzados, etc.

2.5. Modelo TCP/IP

Se conoce con el nombre de modelo de Internet. Define 4 categorías de funciones que deben tener lugar para que las comunicaciones sean exitosas. La arquitectura de la suite de protocolos TCP/IP sigue la estructura de este modelo. Por esto, es común que al modelo de Internet se lo conozca como modelo TCP/IP.

La mayoría de los modelos de protocolos describen un stack de protocolos específicos del proveedor. Sin embargo, puesto que el modelo TCP/IP es un estándar abierto (Figura II.2), una compañía no controla la definición del modelo. Las definiciones del estándar y los protocolos TCP/IP se explican en un foro público y se definen en un conjunto de documentos disponibles al público. Estos documentos se denominan Solicitudes de comentarios (RFCs).



Figura II.2. Capas del modelo TCP / IP

2.6. Protocolos más comunes.

2.6.1. ARP, Address Resolution Protocol

El protocolo ARP tiene un papel clave entre los protocolos de capa de Internet relacionados con el protocolo TCP/IP, ya que permite que se conozca la dirección física de una tarjeta de interfaz de red correspondiente a una dirección IP. Por eso se llama Protocolo de Resolución de Dirección (Figura II.3). Cada equipo conectado a la red tiene un número de identificación de 48 bits. Éste es un número único establecido en la fábrica en el momento de fabricación de la tarjeta. Sin embargo, la comunicación en Internet no utiliza directamente este número, sino que utiliza una dirección lógica asignada por un organismo: la dirección IP.

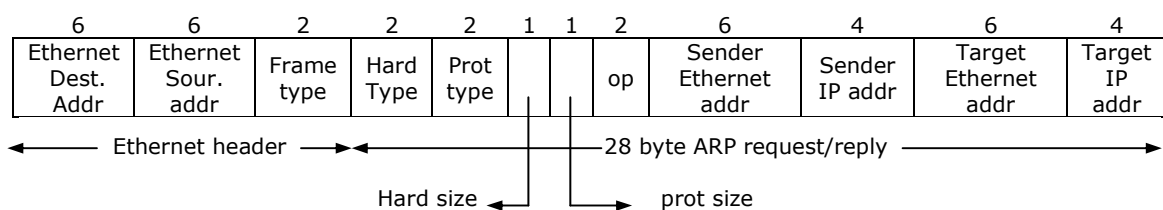


Figura II.3. Formato de paquete ARP de solicitud o reenvío con Ethernet.

- **Direcciones Ethernet origen y destino.** La dirección especial de destino Ethernet con todos los 48 bits puestos a uno (0xFFFFFFFFFFFF) significa una dirección broadcast.
- **Tipo de Frame de Ethernet.** Especifica el tipo de datos que tiene. Por ejemplo, una solicitud ARP pregunta por la dirección de hardware correspondiente a una dirección de protocolo.
- **Tipo de Hardware (Hard).** Especifica el tipo de dirección de hardware. Su valor es 1 para un Ethernet.
- **Tipo de Protocolo (Prot).** Especifica el tipo de dirección de protocolo que ha sido mapeada. Su valor es 0x0800 para una dirección IP.
- **Tamaño de Hardware y Protocolo.** Especifica el tamaño (en bytes) de la dirección de hardware y de la dirección de protocolo.
- **Operación (Op).** Especifica si la operación es una solicitud ARP (un valor de 1, reenvío ARP (2), solicitud RARP (3), reenvío (4). Este campo se requiere desde que los campos de tipo de frame son los mismos para solicitud ARP y para reenvío ARP.

2.6.2. IP, Internet Protocol.

Permite enviar bloques de datos, llamados datagramas, desde un host a otro, estos hosts son identificados por una dirección de longitud fija. IP también provee capacidades de fragmentación y reensamblado de paquetes si es

necesaria la transmisión por redes cuyo tamaño de paquete máximo sea muy pequeño.

El protocolo de Internet trabaja principalmente con direcciones. Las direcciones están compuestas por 4 octetos conformando un número de 32 bits, donde los primeros números indican el número de la red, seguidos por el número local dentro de dicha red. El datagrama IP esta compuesto por los siguientes campos que se muestran en la Figura II.4.

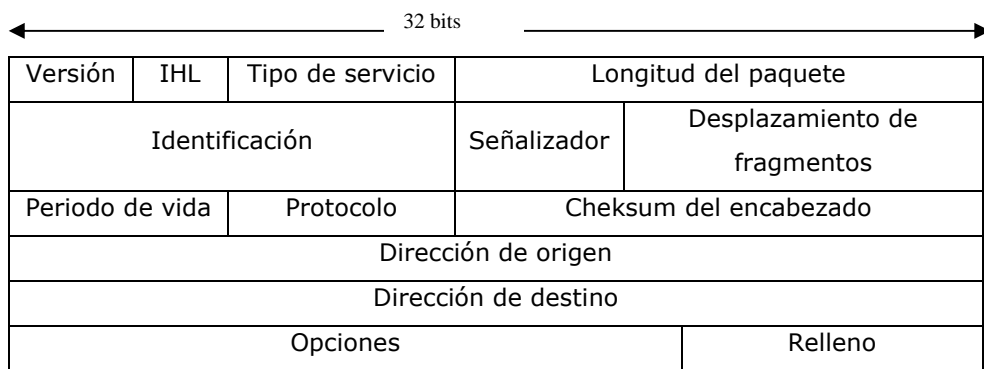


Figura II.4. Cabecera de un datagrama IP

- **Versión:** La versión actual del protocolo es 4, aunque está en experimento la versión 6 pero que no se encuentra muy difundida, por esta razón en el presente trabajo solo se considerará de aquí en adelante a la versión 4 como única. La versión 4 es llamada IPv4.
- **Longitud de la cabecera:** Es el número de palabra de 32-bit en la cabecera, incluyendo cualquier opción. Esto limita la cabecera a 60 bytes.
- **Tipo de servicio:** El campo (TOS) es compuesto por un campo de precedencia de 32-bits, 4 bits de TOS, y un bit no usado que debe ser 0.

- **Longitud Total:** Es la longitud total del datagrama IP en bytes. Usando este campo y el campo de la longitud de la cabecera, se conoce donde la porción de datos del data grama IP inicia y su longitud. Desde que este es un campo de 16-bit, el tamaño máximo de un datagrama IP es de 65535 bytes.
- **Identificación:** Singularmente identifica cada datagrama enviado por un host. Normalmente incrementa por uno cada vez que un datagrama se envía.
- **Tiempo de vida: Define** un límite superior en el número de routers que el datagrama puede pasar. Cuando este campo alcanza el 0, el datagrama es descartado, y el host origen es notificado con un mensaje ICMP. Esto evita que un paquete quede retenido en lazos infinitos de ruteo.
- **Protocolo:** Es usado por IP para demultiplexar los datagramas entrantes. Identifica que protocolo entregó los datos para IP enviarlo.
- **Checksum de la cabecera:** Es solamente calculado sobre la cabecera de IP. Cada datagrama IP contiene la dirección IP origen y la dirección IP destino, este es un valor de 32-bit.
- **Campo de Opción:** Es una lista de longitud variable de información opcional para los datagramas.

2.6.3. ICMP, Internet Control Message Protocol

El mensaje ICMP utiliza la cabecera básica de 20 bytes del datagrama IP y en el primer byte del campo de datos del datagrama IP se encuentra a su vez el campo Tipo de mensaje ICMP. Dependiendo del valor de este campo Tipo, será la

organización de los datos siguientes. Dos mensajes ICMP de importancia son el mensaje de tipo ICMP-ECHO-REQUEST y el mensaje de tipo ICMP-ECHO-REPLY (Figura II.5). El primero es enviado por un host hacia un host destino, constando de una secuencia determinada de bytes. Si host destino lo recibe, enviara entonces el segundo tipo de mensaje, manteniendo la misma secuencia de bytes.

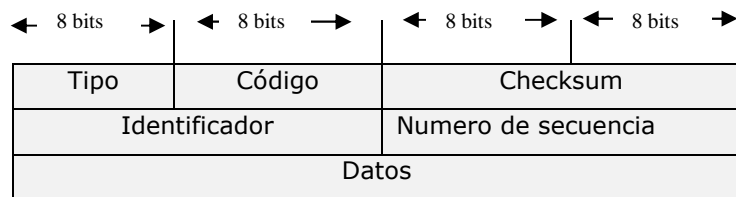


Figura II.5. Mensaje ICMP Echo Request y Echo Reply

Los mensajes ICMP que se pueden enviar incluyen:

- Confirmación de host
- Destino o servicio inalcanzable
- Tiempo excedido
- Redirección de ruta
- Disminución de velocidad en origen

2.6.3.1. Comando Ping

Es una utilidad para probar la conectividad IP entre hosts. Ping envía solicitudes de respuestas desde una dirección host específica. Ping usa un protocolo de capa 3 que forma parte del conjunto de aplicaciones TCP/IP como es ICMP.

Si el host en la dirección especificada recibe la solicitud de eco, éste responde con un datagrama de respuesta de eco ICMP. En cada paquete enviado, el ping mide el tiempo requerido para la respuesta. A medida que se recibe cada respuesta, el ping muestra el tiempo entre el envío del ping y la recepción de la respuesta. Ésta es una medida del rendimiento de la red. Ping posee un valor de límite de tiempo de espera para la respuesta. Si no se recibe una respuesta dentro de ese intervalo

de tiempo, el ping abandona la comunicación y proporciona un mensaje que indica que no se recibió una respuesta.

Después de enviar todas las peticiones, la utilidad de ping provee un resumen de las respuestas. Este resumen incluye la tasa de éxito y el tiempo promedio del recorrido de ida y vuelta al destino.

Los tipos de mensajes ICMP, y los motivos por los que se envían, son vastos. Se tratarán algunos de los mensajes más comunes.

2.6.4. TCP, Transfer Control Protocol

Es el protocolo de transporte orientado a conexión. Entre las funciones especificadas por TCP son entrega confiable y de control de flujo. Cada segmento de TCP posee 20 bytes de carga en el encabezado, que encapsulan los datos de la capa de Aplicación. Las conexiones que utilizan TCP son: Exploradores Web, e-mail, Transferencia de archivos.

Cuando dos hosts se comunican utilizando TCP, se establece una conexión antes de que puedan intercambiarse los datos. Luego de que se completa la comunicación, se cierran las sesiones y la conexión finaliza. Los mecanismos de conexión y de sesión habilitan la función de confiabilidad de TCP.

2.6.4.1. Encabezado TCP

El host rastrea cada segmento de datos dentro de una sesión e intercambia información sobre los datos recibidos por cada host a través de la información del encabezado TCP (Figura II.6).

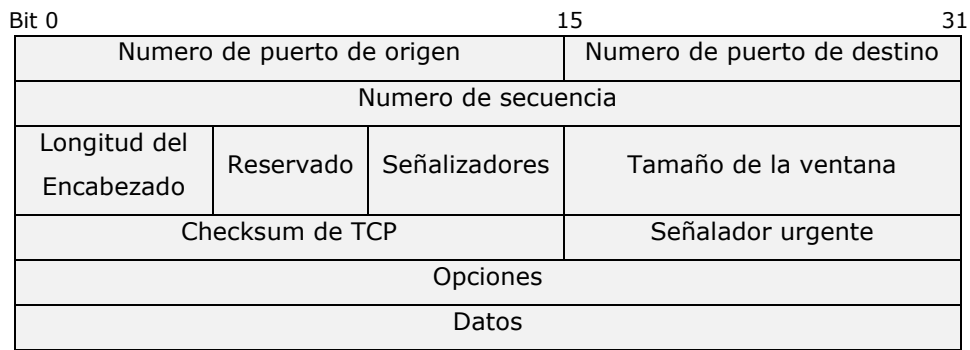


Figura II.6. Campos del encabezado TCP

- **Numero de puerto de origen y destino:** Cada segmento TCP contiene el número de Puerto origen y destino para identificar la aplicación de envío y recepción.
- **Número de Secuencia:** Identifica el stream de bytes de datos del transmisor TCP al receptor TCP. TCP numera cada byte con un número de secuencia.
- **Número de Reconocimiento (Acknowledge):** Contiene el siguiente número de secuencia que el transmisor del reconocimiento espera recibir. Este es por consiguiente el número de secuencia más 1 del último byte de datos recibido exitosamente.
- **Longitud de la Cabecera:** TCP es limitado a una cabecera de 60-bytes. Sin opciones, sin embargo, el tamaño normal de 20 bytes.
- **Bits de Banderas:** Hay seis bits de banderas en la cabecera TCP. Una o más de estas pueden ser cambiadas al mismo tiempo.
- **Tamaño de la Ventana:** El control de flujo de TCP es provisto por cada host advirtiendo un tamaño de ventana. Este es un campo de 16-bit, limitado por una ventana de 65535 bytes.

- **Checksum:** Cubre al segmento TCP: la cabecera y datos. Este es un campo obligatorio y debe ser calculado y almacenado por el transmisor, y verificado por el receptor.

- **El puntero Urgent:** El puntero urgent es valido solamente si la bandera URG está puesta a uno.

- **Checksum:** La opción más común es el campo MSS, tamaño máximo del segmento (Maximum Segment Size). Cada host en una conexión normalmente especifica su opción en el primer segmento intercambiado.

- **Datos:** La porción de datos de la cabecera TCP es opcional. Una cabecera sin datos es usada para recepción de datos de reconocimiento, si no hay datos a ser transmitidos en esa dirección.

2.6.4.2. Establecimiento de la conexión TCP

Para establecer la conexión, los hosts realizan un intercambio de señales de tres vías (Figura II.7). Los bits de control en el encabezado TCP indican el progreso y estado de la conexión.

En conexiones TCP, el host que brinde el servicio como cliente inicia la sesión al servidor. Los tres pasos para el establecimiento de una conexión TCP son:

1. El cliente que inicia la conexión envía un segmento que contiene un valor de secuencia inicial, que actúa como solicitud para el servidor para comenzar una sesión de comunicación.
2. El servidor responde con un segmento que contiene un valor de reconocimiento igual al valor de secuencia recibido más 1, además de su propio valor de secuencia de sincronización. El valor es uno mayor que el

número de secuencia porque el ACK es siempre el próximo Byte u Octeto esperado. Este valor de reconocimiento permite al cliente unir la respuesta al segmento original que fue enviado al servidor.

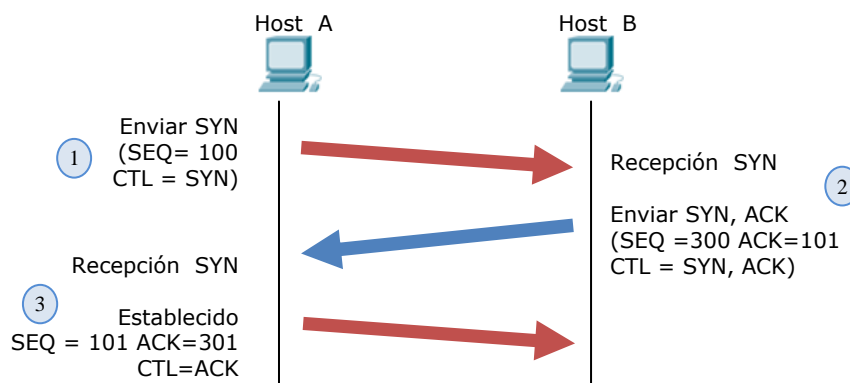


Figura II.7. Proceso conexión TCP

3. El cliente que inicia la conexión responde con un valor de reconocimiento igual al valor de secuencia que recibió más uno. Esto completa el proceso de establecimiento de la conexión.

Para entender el proceso de enlace de tres vías, es importante observar los distintos valores que intercambian los dos hosts. Dentro del encabezado del segmento TCP, existen seis campos de 1 bit que contienen información de control utilizada para gestionar los procesos de TCP. Estos campos son los siguientes:

URG: Urgente campo de señalizador significativo,

ACK: Campo significativo de acuse de recibo,

PSH: Función de empuje,

RST: Reconfiguración de la conexión,

SYN: Sincronizar números de secuencia,

FIN: No hay más datos desde el emisor.

2.6.5. HTTP, Hypertext Transfer Protocol

Es un protocolo común que regula la forma en que interactúan un servidor Web y un cliente Web (Figura II.8). HTTP define el contenido y el formato de las solicitudes y respuestas intercambiadas entre el cliente y el servidor. Tanto el cliente como el software del servidor Web implementan el HTTP como parte de la aplicación. El protocolo HTTP se basa en otros protocolos para regir de qué manera se transportan los mensajes entre el cliente y el servidor

2.6.5.1. Transacción HTTP

La comunicación entre el navegador y el servidor se lleva a cabo en dos etapas:

- El navegador realiza una solicitud HTTP
- El servidor procesa la solicitud y después envía una respuesta HTTP

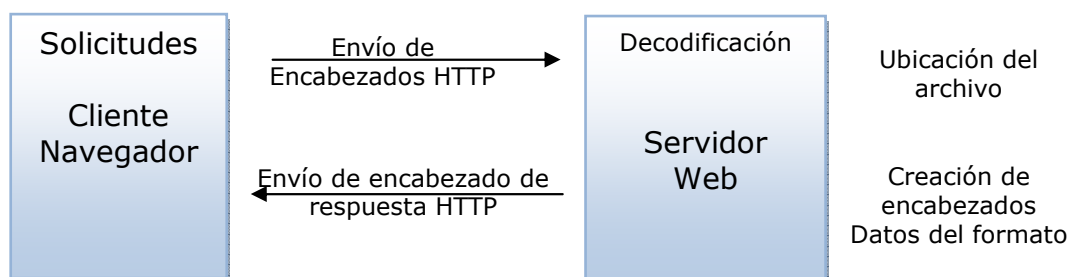


Figura II.8. Diálogo HTTP

2.6.5.2. Solicitud HTTP

El cliente realiza la **solicitud HTTP**, hacia un host, puerto 80 que es el puerto por defecto para http. El mensaje se envía en el estilo siguiente:

```
GET / index.html/ HTTP1.1
Host: www.example.com
User-agent: nombre-cliente
[Línea en blanco]
```

[Cuerpo de la solicitud]

2.6.5.3. Respuesta HTTP

El servidor procesa la solicitud y después envía una **respuesta http**, en el caso de una página Web el mensaje sería:

```
http/1.1 200 OK
Date: Fri, 31 Dec 2008 23:59:59 GMT
Content-Type: text/html
Content_Length: 1221
<html>
<body>
<h1>Pagina principal de ...</h1>
(Contenido)
</body>
</html>
```

2.6.5.4. Métodos HTTP

La primera palabra en la línea de solicitud es el nombre del método a ser ejecutado. Desde que el método es especificado en esta forma, HTTP puede ser expandido para cubrir las necesidades del futuro para aplicaciones orientadas a objetos. Los métodos comunes están listados en la Tabla II.1

Tabla II.1. Métodos en la solicitud http

Comando	Descripción
GET	Solicita el recurso ubicado en la URL especificada
HEAD	Solicita el encabezado del recurso ubicado en la URL especificada
POST	Envía datos al programa ubicado en la URL especificada
PUT	Envía datos a la URL especificada
DELETE	Borra el recurso ubicado en la URL especificada

2.7. Microcontroladores

Desde la invención de los semiconductores, el desarrollo de la tecnología digital ha dado lugar a dispositivos cada vez más complejos y rápidos. Entre ellos los microprocesadores y los microcontroladores. Los microcontroladores se encuentran en todo aparato electrónico que posea un grado de automatismo.

Desde comienzos de los 90 todos los elementos del controlador se han podido incluir en un solo circuito integrado. Un microcontrolador dispone normalmente de los siguientes componentes:

- Procesador o CPU (Unidad Central de Proceso).
- Memoria RAM para contener los datos.
- Memoria para el programa tipo ROM/EPROM/EEPROM/Flash.
- Líneas de E/S para comunicarse con el exterior.

También puede poseer otros bloques de apoyo que flexibilizan aún más su uso, tales como:

- Módulos para el control de periféricos: temporizadores, puertos serie y paralelo, CAD: Conversores Analógico/Digital, etc.
- Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.
- Sistemas de protección de sobrecorriente o cortocircuito

2.7.1. Arquitecturas de proceso

En general, los microcontroladores poseen dos formas de trabajo en cuanto a los datos y direcciones:

- La arquitectura de Von Neuman se caracteriza por disponer de una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. A dicha memoria se accede a través de un sistema de buses único.
- La arquitectura Harvard dispone de dos memorias independientes, una que contiene sólo instrucciones y otra sólo datos. Ambas disponen de sus respectivos sistemas de buses de acceso y es posible realizar operaciones de acceso simultáneamente en ambas memorias.

2.7.2. La unidad central de proceso O CPU

Se encarga de direccionar la memoria de instrucciones, su decodificación y la ejecución de la operación que implica la instrucción, así como la búsqueda de los operandos y el almacenamiento del resultado. Existen tres tipos de CPU en cuanto a la forma de "procesar" las instrucciones:

- **CISC** (Computadores de Juego de Instrucciones Complejo): Disponen de más de 80 instrucciones de máquina en su repertorio, algunas de las cuales son potentes, requiriendo muchos ciclos para su ejecución.
- **RISC** (Computadores de Juego de Instrucciones Reducido): El repertorio de instrucciones de máquina es muy reducido y las instrucciones son simples y, generalmente, se ejecutan en un ciclo. La sencillez y rapidez de las instrucciones permiten optimizar el hardware y software del procesador.
- **SISC** (Computadores de Juego de Instrucciones Específico): El juego de instrucciones, además de ser reducido, es "específico", es decir, las instrucciones se adaptan a las necesidades de la aplicación prevista.

2.7.3. Memorias

Existen distintos tipos de memorias y en un microcontrolador se puede encontrar cualquiera de ellas:

- **ROM** con máscara: Es una memoria no volátil de sólo lectura, cuyo contenido se graba durante la fabricación del chip. El término máscara viene de la forma como se fabrican los circuitos integrados.
- **OTP**: One Time Programmable, es una memoria no volátil de sólo lectura "programable una sola vez" por el usuario. Se puede escribir el programa en el chip mediante un sencillo grabador controlado por un software desde una PC.
- **EPROM**: Erasable Programmable Read Only Memory, pueden borrarse y grabarse muchas veces. La grabación se realiza, como en el caso de los OTP, con un grabador gobernado desde una PC. Hoy día se utilizan poco, siendo sustituidas por memorias EEPROM o Flash.
- **EEPROM**: Electrical Erasable Programmable Read Only Memory, son memorias de sólo lectura, programables y borrables. El número de veces que puede grabarse y borrarse una memoria EEPROM es finito, por lo que no es recomendable una reprogramación continúa.

2.7.4. Puertos de Entrada y Salida

Un microcontrolador debe soportar líneas de E/S que comunican al computador interno con los periféricos exteriores. Por lo general, estas líneas se agrupan de ocho en ocho formando Puertos. Las líneas digitales de los Puertos pueden configurarse como Entrada o como Salida, cargando un 1 ó un 0 en el bit correspondiente de un registro destinado a su configuración.

2.7.5. Recursos especiales del microcontrolador

Cada fabricante posee numerosas versiones de una arquitectura básica de microcontrolador. En algunas familias se amplía las capacidades de las memorias, en otras se incorporan nuevos recursos, en otras se reduce las prestaciones al mínimo para aplicaciones muy simples, etc.

- **Modulador de Ancho de Pulsos o PWM.-** Son circuitos que proporcionan en su salida impulsos de anchura variable, que se ofrecen al exterior a través de las patillas del encapsulado. Es útil para sistemas de control de potencia, como por ejemplo motores.

- **Puertos de Comunicación.-** Para que el micro se pueda comunicar con otros dispositivos, otros buses de microprocesadores, buses de sistemas, buses de redes y poder adaptarlos con otros elementos bajo otras normas y protocolos es preciso agregarle unidades o puertos de comunicación. Algunos modelos disponen de recursos que permiten esta tarea, entre los que destacan:
 - UART, adaptador de comunicación serie asincrónica.
 - USART, adaptador de comunicación serie sincrónica y asincrónica.
 - USB (Universal Serial Bus). Bus I2C, que es una interfaz serie de dos hilos desarrollado por Philips.
 - Interface SPI, un puerto serie sincrónico.
 - CAN (Controller Area Network), para permitir la adaptación con redes de conexionado multiplexado desarrollado conjuntamente por Bosch e Intel para el cableado de dispositivos en automóviles. En EE.UU. se usa el J1850.

2.8. SPI (Serial Peripheral Interface)

El protocolo SPI es un bus de tres líneas, sobre el cual se transmiten paquetes de información de 8 bits. Dos líneas transfieren los datos (una en cada dirección) y la tercer línea es la del reloj.

Los dispositivos conectados al bus son definidos como maestros y esclavos (Figura II.9). Un maestro es aquel que inicia la transferencia de información sobre el bus y genera las señales de reloj y control. Un esclavo es un dispositivo controlado por el maestro. Cada esclavo es controlado sobre el bus a través de una línea selectora llamada Chip Select o Select Slave, por lo tanto el esclavo es activado solo cuando esta línea es seleccionada. El bus SPI emplea un simple registro de desplazamiento para transmitir la información.

2.8.1. Especificaciones del bus

Todas las líneas del bus transmiten la información sobre una sola dirección.

- La señal sobre la línea de reloj (**SCLK**) es generada por el maestro y sincroniza la transferencia de datos.
- La línea **MOSI** (Master Out Slave In) transporta los datos del maestro hacia el esclavo.
- La línea **MISO** (Master In Slave Out) transporta los datos del esclavo hacia el maestro.
- Cada esclavo es seleccionado por un nivel lógico bajo a través de la línea (**CS** = Chip Select o **SS** Slave Select). Los datos son transferidos en bloques de 8 bits, en donde el bits más significativo (MSB) se transmite primero.

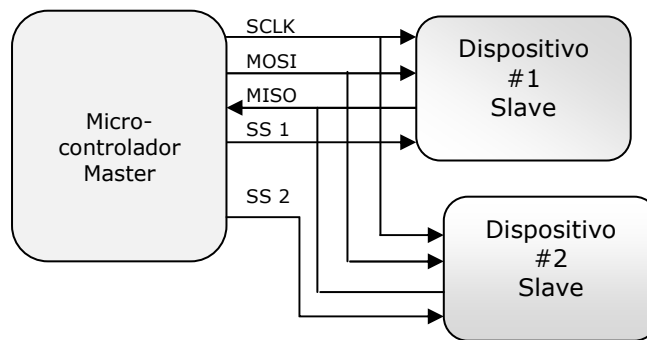


Figura II.9. Asignación de líneas Maestro - Esclavo

2.8.2. Ventajas

- Comunicación full-duplex
- Mayor velocidad de transmisión que con(I²C) o SMBus
- No está limitado a la transferencia de bloques de 8 bits.
- Elección del tamaño de la trama de bits, de su significado y propósito.
- Consume menos energía que I²C o que SMBus debido que posee menos circuitos y estos son más simples.
- No es necesario arbitraje o mecanismo de respuesta ante fallos.
- Los dispositivos esclavos usan el reloj que envía el master, no necesitan por tanto su propio reloj.
- Como mucho una única señal específica para cada esclavos (señal SS), las demás señales pueden ser compartidas

2.8.3. Desventajas

- Consume más terminales en cada chip que I²C.
- El direccionamiento se hace mediante líneas específicas (señalización fuera de banda) a diferencia de lo que ocurre en I²C que se selecciona cada chip mediante una dirección de 7 bits que se envía por las mismas líneas del bus.

- No hay control de flujo por hardware.
- No permite fácilmente tener varios maestros conectados al bus.

2.9. Modulo LCD

La razón primaria al realizar esta elección es simple, utilizar displays de segmentos exige en la mayoría de los casos utilizar técnicas de multiplexación en el microcontrolador, de forma de reducir líneas de entrada / salida aplicadas al manejo de los displays porque de otra forma estos se necesitarían de 8 pines de salida por cada dígito. En aplicaciones donde sea necesario mostrar valores que exceden los 4 dígitos, conviene utilizar un display LCD.

Las técnicas de multiplexación permiten manejar el LCD con menos pines, consume menos energía que un sistema de displays de segmentos, pero se pierden numerosos recursos de memoria para manejar la multiplexación dentro del microcontrolador, además de "robar tiempo" al programa principal.

2.9.1. Tipos de displays LCD

En el mercado hay una gran variedad de LCDs (Figura II.10). El display LCD que se seleccionó es del tipo alfanumérico, de 2 líneas y 16 caracteres por línea. Es uno de los más utilizados y populares. También se fabrican displays de 1, 2, 3 y 4 líneas por 8 caracteres, 16 caracteres, 20 caracteres y también 40 caracteres por línea.

Por cada modelo varían los colores del display y pueden disponer de retroiluminador incorporado o no, dependiendo esta elección del lugar donde se instale el dispositivo que lo contenga y si hay luz natural o no.



Figura II.10. Tipos de LCD

Otra de las diferencias que pueden tener entre ellos es la forma de la matriz de puntos con la cual conforman los caracteres. Existen de 5 columnas por 7 filas y de 5 columnas por 8 filas. Este último puede utilizar el cursor en la última fila de cada carácter.

2.10. Sensores

Un sensor es un dispositivo eléctrico y/o mecánico que convierte magnitudes físicas (luz, magnetismo, presión, etc.) en valores medibles de dicha magnitud como puede ser voltaje o frecuencia. No existe una tecnología de medición que sea apropiada para todas las aplicaciones. El mercado ofrece varios tipos sensores que se utilizaran según su finalidad.

2.10.1. Tipos de Sensores

Existe una gran cantidad de sensores en el mercado, para poder medir magnitudes físicas, de los que se pueden enumerar los siguientes:



- Temperatura
- Humedad
- Presión

- Posición
- Movimiento
- Acústicos, etc.

Existen dispositivos que miden 2 magnitudes a la vez como es el caso de los presentados en la Tabla II.2, que miden temperatura y humedad.

Ante la necesidad de medir la temperatura y la humedad para este proyecto a continuación se describirán varias alternativas.

TABLA II.2. Sensores de Humedad y Temperatura

Nombre	Rango de Temperatura	Rango de humedad	Salida	Empaque
SHT11	-40 a 125°C	0-100RH	Digital (interface serial)	
CHIPCAP D	-50 a 150°C	10-95 RH	Digital Manchester 8 bits	

2.10.2. Sensor de Temperatura

2.10.2.1. Temperatura

Un cuerpo caliente tiene más energía térmica que un cuerpo "frío" semejante. Esto significa que, las partículas del cuerpo con mayor temperatura tienen más energía cinética y potencial que las partículas del cuerpo "frío".






La cantidad de calor o el grado de calor se mide sobre una escala específica, es una propiedad de un cuerpo llamada temperatura. La temperatura nos indica hacia dónde va a fluir el calor cuando se ponen dos cuerpos en interacción; el calor pasa

del cuerpo de mayor temperatura al de menor temperatura, hasta que ambos lleguen a una temperatura estable.

2.10.2.2. Tipos de Sensores de Temperatura

Existe gran variedad de tipos de sensores de temperatura como: Termopares, Resistivos y Semiconductores, que se utilizan de acuerdo a la aplicación que se esta realizando. En la Tabla II.3 se presenta los sensores de temperatura que existen en el mercado:

TABLA II.3. Sensores de temperatura en el mercado

Nombre	Rango de temperatura	Precisión	Salida	Empaque
Termistor	-45 a 125 °C	$\pm 0.3^{\circ}\text{C}$	Analógica	
Lm335az	-40 a 100 °C	10 mv/K ^o	Analógica	
Lm35	-55 a 150 °C	1 °C	Analógica, salida lineal	
DS18B20	-10 a 85 °C	$\pm 0.5^{\circ}\text{C}$ 9 bits	Digital interface 1 - wire	
TMP100A	-55 a 125 °C	9 a 12 bits	Digital Protocolo I2C	

De los sensores anteriormente expuestos se seleccionó el termistor de 10K (Anexo 1), por ser económico y popular en el mercado, además el termistor resulta muy adecuado para mediciones precisas de temperatura, se utiliza ampliamente para aplicaciones de control. Los termistores sirven para la medición o detección de temperatura tanto en gases, como en líquidos o sólidos.

2.10.2.3. Aplicaciones de los Sensores de Temperatura

- Control de medidas de temperatura como en robots bomberos.
- Monitoreo y control en calderas.
- Detección de sobrecalentamiento en sistemas de control etc.

2.10.3. Sensor de humedad

El alcance que la influencia de la humedad podría tener en cualquier proceso industrial puede variar pero es esencial que al menos sea monitoreada, y en muchos casos controlada. La humedad es una propiedad más difícil de definir y medir que sus parámetros asociados como pueden ser la presión y temperatura.



2.10.3.1. Humedad Relativa

La medición de la humedad relativa consiste en la relación entre la presión parcial del vapor de agua en el gas de que se trate y la presión de saturación del vapor, a una temperatura dada. Por lo tanto la humedad relativa es función de la temperatura. La medición es expresada como un porcentaje. La humedad relativa es un parámetro utilizado principalmente en aplicaciones ambientales (acondicionamiento de aire) o mediciones meteorológicas ya que impacta directamente en el confort humano. Cuando los niveles de humedad relativa son bajos puede producirse electricidad estática (cuando la RH esta entre 30 y 35%), permitiendo que se dañe el equipamiento electrónico.

2.10.3.2. Tipos de Sensores de Temperatura

En el mercado existen diferentes tipos de sensores de temperatura, en la tabla II.4 detallaremos 2 de ellos.

TABLA II.4. Sensores De Humedad en el mercado

Nombre	Temp. de operación	Rango de humedad %	Tiempo de respuesta	Empaque
HS1101	-40 a 100°C	0 – 100 RH	5 s Salida digital	
HIH4000A	-40 a 85°C	10 – 95 RH	15 s Salida analógica	

Se decidió utilizar el sensor HS1101 (Anexo 2) por tener disponibilidad en nuestro país y tener el mejor tiempo de respuesta.

2.10.3.3. Efectos de la temperatura y la humedad

La salida de todos los sensores de humedad por absorción (capacitivos, resistivos, etc.), se ven afectados sensiblemente por la temperatura. A causa de esto se utilizan mecanismos de compensación de temperatura en aplicaciones que demanden alto nivel de precisión o un amplio rango de temperaturas. Cuando se compensa la temperatura de un sensor lo mejor es hacer la medición de temperatura tan cerca como sea posible de área activa del sensor, esto es en el mismo micro-ambiente.

2.10.3.4. Aplicaciones de los Sensores de Humedad

- Son ampliamente usados para sistemas de control agrícolas o agroindustriales en donde la humedad es una variable importante para controlar
- En sistemas de almacenamiento
- Automatización oficinas, control de aire acondicionado
- Estaciones de clima, etc.

2.11. Actuadores

Son todos los elementos que pueden provocar un efecto sobre un proceso automatizado. Los actuadores son dispositivos capaces de generar una fuerza a partir de líquidos, de energía eléctrica y gaseosa. El actuador recibe la orden de un controlador y da una salida necesaria para activar a un elemento final de control.

Existen tres tipos de actuadores:

- **Hidráulicos.**- Por lo general, los actuadores hidráulicos se emplean cuando se necesita potencia, sin embargo, los hidráulicos requieren mucho equipo para suministro de energía.
- **Neumáticos.**- las aplicaciones de los modelos neumáticos también son limitadas desde el punto de vista de precisión y mantenimiento
- **Eléctricos.**- Los actuadores eléctricos también son muy utilizados en los aparatos mecatrónicos, como por ejemplo, en los robots. Los servomotores CA (Figura II.11) sin escobillas se utilizarán en el futuro como actuadores de posicionamiento preciso debido a la demanda de funcionamiento sin tantas horas de mantenimiento.



Figura II.11. Actuador Eléctrico: Servomotor

Los actuadores más usuales son:

- **Cilindros neumáticos e hidráulicos.** Realizan movimientos lineales.
- **Motores (actuadores de giro) neumáticos e hidráulicos.** Realizan movimientos de giro por medio de energía hidráulica o neumática.
- **Válvulas.** Las hay de mando directo, motorizadas, electroneumáticas, etc. Se emplean para regular el caudal de gases y líquidos.
- **Resistencias calefactoras.** Se emplean para calentar.
- **Motores eléctricos.** Los más usados son de inducción, de continua, sin escobillas y paso a paso.
- **Bombas, compresores y ventiladores.** Movidos generalmente por motores eléctricos de inducción.

Los actuadores en el presente proyecto estarán orientados a mantener y controlar la temperatura y humedad.

2.11.1. Ventilación

El aire debe ser renovado cada cierto tiempo, para ello se ha dispuesto ventiladores que cumplirán con la función de introducir y sacar aire para regular las condiciones de humedad y temperatura según convenga. El ventilador (Figura II.12) se utiliza para desplazar aire o gas de un lugar a otro, dentro de o entre espacios, para motivos industriales o uso residencial, para ventilación o para aumentar la circulación de aire en un espacio habitado, básicamente para refrescar. Por esta razón, es un elemento indispensable en climas cálidos



Figura II.12. Ventilador de 12V y 12Cm

2.11.2. Calefacción

Son dispositivos que emiten calor a través del cual calientan el aire. Existen diferentes tipos como: los sistemas de calefacción Radiantes que distribuyen temperaturas en un lugar de manera uniforme, caloveiladores eléctricos (Figura II.13) que internamente tienen resistencias eléctricas de entrega de calor instantáneo, a gas que solo se recomienda en lugares donde exista ventilación, debido al monóxido de carbono que emana, entre otros.



Figura II.13. Caloveilador eléctrico

CAPITULO III

ESTUDIO DE LOS SISTEMAS WEB EMBEBIDOS

3.1. Sistemas embebidos

Se puede definir al sistema embebido (SE) como un subsistema electrónico de procesamiento, programado para realizar una o pocas funciones para cumplir con un objetivo específico. Generalmente es parte integral de un sistema mayor.

Por el contrario, un sistema de procesamiento de propósito general, tal cual una computadora personal, puede realizar diferentes tareas dependiendo de la programación.

3.1.1. Partes de un SE

Un sistema embebido sigue un modelo (Figura III.14) que está integrado por hardware y el software embebido o empotrado como parte esencial del mismo sistema, conocido en inglés como firmware o embedded software.

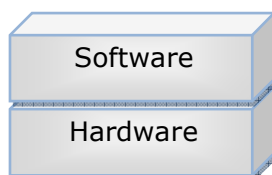


Figura III.14. Modelo de un Sistema Embebido

- El hardware lo constituye un microcontrolador, que es el corazón del sistema y de otros componentes electrónicos de acuerdo a la aplicación que se está implementando como por ejemplo: componentes digitales, analógicos, bloques de lógica combinatoria, etc.
- El software del sistema es el encargado de hacer que los distintos componentes de éste realicen las funciones en los tiempos definidos y de acuerdo a sus requerimientos. Este software está incorporado en el microcontrolador e interactúa con los otros componentes a través de buses de datos, de control y puertos.

3.1.2. Ventajas al utilizar SE

- Debido a que un sistema embebido está dedicado a una tarea específica, el diseño puede ser optimizado reduciendo los costos, el tamaño del producto y el consumo de potencia, a la vez de incrementar la confiabilidad y eficiencia.
- Los sistemas embebidos se pueden utilizar en sistemas de tiempo real o sistema rápido todo dependerá de los requerimientos específicos de la aplicación. De igual forma, el sistema será rápido si y sólo si se requiere de esta rapidez de acuerdo a la definición específica del sistema

3.1.3. Desventajas de los SE

- Un sistema basado en este tipo de tecnología se ejecuta en un hardware limitado tanto en velocidad como en cantidad de memoria.
- El software debe ser extremadamente confiable, muy eficiente y compacto, y muy preciso en su respuesta en el instante de la transmisión de la información de ingresos y salidas. Su tolerancia a fallas es muy baja, porque una vez en manos del usuario en la mayoría de los casos es muy difícil o imposible de realizar cambios.
- Requiere de mano de obra sumamente especializada, no sólo de carácter informático y electrónico sino de otra con conocimientos profundos del funcionamiento de los diferentes elementos que conforman el entorno en el que debe funcionar el sistema embebido.

3.1.4. Aplicaciones

Este tipo de dispositivos son utilizados en diferentes áreas. Entre las aplicaciones de los sistemas embebidos se expone el siguiente listado:

- En diversos dispositivos frecuentemente en equipamiento de redes de datos como routers, firewalls, switches
- En sistemas de comunicaciones como teléfonos celulares, PDAs, cámaras digitales o para fines hogareños, sistemas de entretenimiento, etc.
- Analizadores electromédicos de parámetros clínicos.
- Conmutadores "inteligentes" para ahorro de energía en iluminación.
- Controladoras industriales de nivel, temperatura, humedad, etc.
- Electroencefalogramas y electrocardiógrafos.
- Equipos "inteligentes" de electrónica de potencia, tales como fuentes ininterrumpibles de energía, máquinas soldadoras, equipos de corte por plasma, de protección catódica, etc.
- Monitores de medio ambiente.

- Monitoreo, programación y control de estudios de radio y televisión.
- Sistemas electrónicos y control en maquinaria agrícola.
- Sistemas de posicionamiento y radioenlace para despacho y control de flota.

3.2. Sistemas Web Embebidos

A veces se hace necesario acceder a cierta información mediante una aplicación Web. Una forma de alcanzar este fin es instalar un servidor Web o de aplicaciones, pero hay otra manera más sencilla y de bajo costo que es un sistema web embebido. Sin embargo no hay que dejar de lado el hecho que los sistemas web embebidos corren en el mismo entorno que nuestra aplicación, compartirán memoria, por lo que no es aconsejable usarlos cuando se esperan muchos usuarios concurrentes o gran carga de trabajo.

Un sistema web embebido al igual que un servidor web típico tiene un programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporcionando los recursos que soliciten usando el protocolo HTTP o el protocolo HTTPS (la versión cifrada y autenticada). Los sistemas web básicos cuentan con un esquema de funcionamiento muy simple, basado en ejecutar infinitamente el siguiente bucle:

1. Espera peticiones en el puerto TCP indicado (el estándar por defecto para HTTP es el 80).
2. Recibe una petición.
3. Busca el recurso.
4. Envía el recurso utilizando la misma conexión por la que recibió petición.
5. Vuelve al segundo punto

Entre las aplicaciones comerciales, se han desarrollado algunos dispositivos que utilizan esta tecnología como es el caso de PICDEM.net y MicroWeb, pero al ser

placas demostrativas su utilización se limita. A continuación se describen estos dispositivos:

- El **PICDEM.net** es una placa demostrativa de Internet/Ethernet usando el microcontrolador PIC16F877 con un firmware TCP/IP. La placa soporta cualquier dispositivo de 40 pines DIP que conforme el estándar de pines del 16F877 o el PIC18C452. Contiene un stack TCP/IP amigable. Un servidor Web con HTML en una memoria serial EEPROM 24L256. Firmware para bajar la página Web a la memoria serial utilizando el protocolo Xmodem. Conector para Interfaz ICSP/ICD, interfaz Ethernet, Interfaz RS232 y módulo LCD 16x2.
- El servidor **MicroWeb** es un servidor experimental basado en una tarjeta de red compatible NE2000 para bus ISA trabajando en modo 8bits, un Microcontrolador PIC16F877 y una memoria 24LS256 I2C para alojar las páginas Web. Cuenta con interfaz RS232, módulo LCD. Implementa ARP, ICMP, IP, TCP y HTTP.

3.3. Selección del Sistema Web Embebido

Para realizar el estudio del sistema se consideró poner énfasis en dos componentes principales como el dispositivo que dará conectividad con la red y el microcontrolador.

3.3.1. Alternativa Ethernet

3.3.1.1. Wiznet W5100

Wiznet oferta al mercado un chip que integra tanto la pila TCP/IP como la MAC por SPI. El W5100 (Figura III.15) trabaja hasta 10Mb/s.

Características principales:

- Soporta los protocolos : TCP, UDP, ICMP, IPv4 ARP, IGMP, PPPoE
- Auto negociación (Full-duplex/half duplex)
- Auto MDI/MDIX
- Soporte de conexión ADSL (con protocolo PPPoE , y autenticación PAP/CHAP)
- Memoria interna de 16Kbytes para los buffers de transmisión/recepción.
- 3.3V Tolerante a 5V en las entradas/salidas
- Empaquetado 80 Pin LQFP



Figura III.15. Modulo Wiznet W5100

3.3.1.2. Packet Whacker

Es un pequeño módulo fabricado por EDTP (Figura III.16), el cual puede agregar conectividad Ethernet a cualquier proyecto. Esta basado en el controlador RLT8019AS. Puede ser usado con la mayoría de los microcontroladores como los AVR de ATMEL y PIC de MICROCHIP y es compatible NE2000. Salida RJ45 y trabaja a 10Mbps/s. La frecuencia de operación es de 20 Mhz



Figura III.16. Packet Whacker

El firmware para el PACKET WHACKER esta disponible en la página oficial del fabricante, incluyendo rutinas simples que permiten el control con los microprocesadores PIC16F877 o PIC18F452. También existe código disponible para los micros de ATMEL vía internet o LAN.

3.3.1.3. Controlador ENC28J60

El ENC28J60 es un controlador Ethernet con un Interfaz periférica serial estándar industrial (Figura III.17). Está diseñado para ser utilizado como una interfaz de red Ethernet para cualquier controlador equipado con SPI. También provee un modulo DMA interno que brinda velocidad a la transmisión de los datos y operaciones de verificación de errores IP apoyadas por el hardware. La comunicación con el controlador del host se implementa a través de dos pines interruptores y del SPI, con un promedio de datos de hasta 10 Mb/s. El chip tiene 28 pines (Figura III.18) y tiene pocos componentes externos, por eso puede ser implantado muy sencillamente en cualquier proyecto.

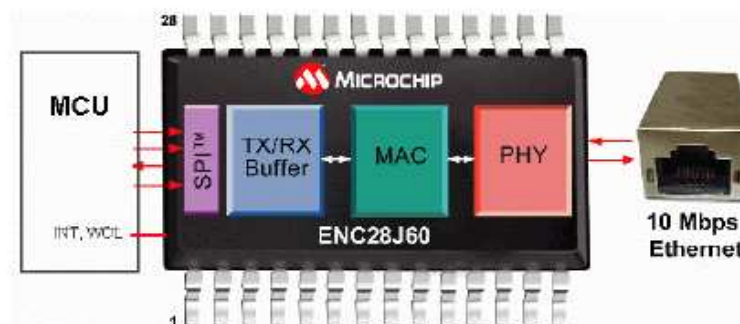


Figura III.17. Controlador ENC28J60

Características del Controlador ENC28j60

- Compatible con IEEE 802.3.
- Soporta la capa MAC y Física con el estándar 10Base-T

- Soporta los modos de comunicación Full y Half-Duplex.
- Retransmisión automática programable cuando hay colisión.
- Interfaz SPI con velocidades de hasta 10Mbits.
- 8-KByte de transmisión/recepción de paquetes de doble puerto SRAM.
- Interior DMA para una rápida circulación de datos.
- Siete fuentes de interrupción, con dos pines de interrupción.
- Reloj de 25MHz.
- Rango de operación: 3, 5 voltios.
- Entradas de nivel TTL.

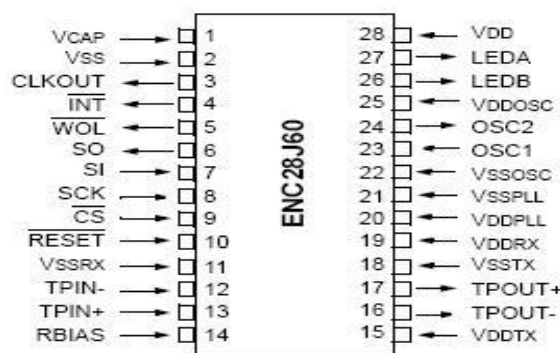


Figura III.18. **ENC28J60**. Encapsulado SPDIP, SSOP, SOIC

3.3.2. Métricas para la elección del módulo Ethernet

Las métricas consideradas para la selección del componente que dará conectividad con la red son:

- **Precio:** Un factor muy importante pues se necesita abaratar costos al máximo posible.
- **Disponibilidad:** En nuestro medio, ¿qué tan difícil es encontrarlo?, se tomó en cuenta la disponibilidad del componente en el país.
- **Facilidad de programación:** Se necesita saber que tan difícil es manejarlo al momento del diseño del firmware.

- **Facilidad de Fabricación del Sistema:** Se debe tomar en cuenta si se necesitan elementos o equipos para la fabricación adicionales a los disponibles.

3.3.3. Estudio de la mejor alternativa

Para la selección del dispositivo Ethernet, se aplicó el Método Ponderado que consiste en dar un valor cuantitativo a las métricas consideradas, en este caso le asignamos valores entre 1 y 10, a continuación los presentamos en la Tabla III.5.

Tabla III.5. Asignación cuantitativa para las métricas

Métricas (Factores)	Valores para Ponderación	Ponderación (V. P/31)
Precio	10	0,32
Disponibilidad	7	0,23
Facilidad de Programación	8	0,26
Facilidad de fabricación del sistema	6	0,19
Total	31	1,00

A continuación registramos en la tabla III.6 los precios, según las páginas Web de empresas de venta de equipos y dispositivos electrónicos:

Tabla III.6. Precios de dispositivos Ethernet

Dispositivo	Precio (\$)
Wiznet (W5100) ¹	20,00
Packet Whacker ²	35,00
ENC28J60 ³	35,00

De acuerdo a los precios y la información técnica presentada anteriormente de cada uno de los dispositivos se determinó una calificación o costo (Tabla III.7) del 1 al 5

¹ Fuente: Página Web Olimex - Chile

² Fuente: Página Web del fabricante EDTP

³ Fuente: Página Web APM – Quito-Ecuador

correspondiente Insuficiente, Regular, Bueno, Muy bueno y Sobresaliente respectivamente. Esta información se obtuvo de las páginas oficiales de los fabricantes.

Tabla III.7. Calificaciones y costos

Factor Alternativa	Precio USD	Disponibilidad	Facilidad de programación	Facilidad de fabricación
Wiznet (W5100)	20,00	1	4	3
Packet Whacker	35,00	1	3	3
ENC28J60	35,00	4	3	4

Posteriormente realizando una matriz de puntos seleccionamos la alternativa adecuada. Esta matriz consta de varios cálculos matemáticos (Tabla III.7).

Para obtener la calificación ponderada se multiplica el valor ya ponderado por la calificación asignada en la tabla anterior (Tabla III.8). Para los valores en dólares se selecciona el valor menor, a ese valor se le asigna la mayor calificación, es decir 10. A continuación se obtienen los excedentes de los demás valores y se aplica una regla de tres. Los procesos mencionados se resumen en la tabla

Tabla III.8. Matriz de puntos

Factores	Ponderación	WIZNET		PACKET		ENC28J60	
		Calif	Calif. P	Calif.	Calif. P	Calif.	Calif. P
Precio	0,32	10	3,23	7,5	2,42	7,5	2,42
Disponibilidad	0,23	1	0,23	1	0,23	5	1,13
Facilidad Programación	0,26	4	1,03	3	0,77	3	0,77
Fabricación del Sistema	0,19	3	0,58	3	0,58	4	0,77
total	1,00	-	5,06	-	4,00	-	5,10

La alternativa que se debe seleccionar es la que tiene mayor costo, en este caso se observa que es: ENC28J60.

3.3.4. Microcontrolador

Después de haber seleccionado el dispositivo Ethernet, es el momento de elegir un dispositivo capaz de controlarlo. El trabajo del diseñador es encontrar el modelo mínimo que satisfaga todos los requerimientos de su aplicación. Así, minimizar el costo, el hardware y el software. En el mercado pueden encontrarse dispositivos capaces de controlar un sistema, FPGA, DSP, Microcontroladores, etc.

Hay una gran variedad de fabricantes de microcontroladores, entre los mas importantes podemos encontrar a Atmel, Hitachi, Intel, Microchip, Motorola, Texas, Zilog, etc. Para la elección del microcontrolador se ha considerado los fabricados por las empresas Microchip y Atmel.

3.3.4.1. Microcontroladores Microchip 16F87X

Ofrece una amplia gama de microcontroladores, su punto fuerte son los microcontroladores de 8 bits, con los que se ha ido imponiendo en el mercado. Recientemente ha presentado una gama nueva de microcontroladores (PIC) y procesadores digitales de señal (DSP) de 16bits con unas capacidades de procesamiento de 40Mips

No se analizaron los PICs de gama alta porque sólo se utilizan en aplicaciones muy especiales con grandes requerimientos. Se optó por la serie 16F87X (Figura III.19) para el análisis, por motivos como la abundante información y herramientas de diseño existente en el mercado. De la serie se seleccionó el PIC16F877A.

También salta a la vista el hecho que es sencillo en el manejo y contiene un buen promedio elevado en los parámetros de velocidad, consumo, tamaño, alimentación.

Las principales características con que cuenta el 16F877A son:

- Procesador de arquitectura RISC avanzada
- Juego de 35 instrucciones con 14 bits de longitud. Todas ellas se ejecutan en un ciclo de instrucción menos las de salto que tardan 2.
- Frecuencia de 20 Mhz
- Hasta 8K palabras de 14 bits para la memoria de código, tipo flash.
- Hasta 368 bytes de memoria de datos RAM
- Hasta 256 bytes de memoria de datos EEPROM
- Hasta 14 fuentes de interrupción internas y externas
- Pila con 8 niveles
- Modos de direccionamiento directo, indirecto y relativo
- Perro guardian (WDT)
- Código de protección programable
- Modo Sleep de bajo consumo
- Programación serie en circuito con 2 patitas
- Voltaje de alimentación comprendido entre 2 y 5.5 voltios
- Bajo consumo (menos de 2mA a 5V y 5Mhz)

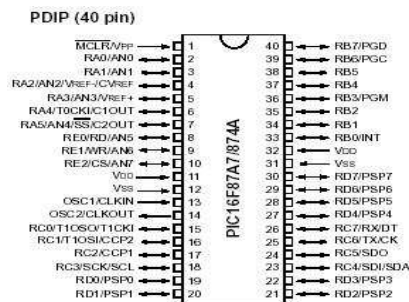


Figura III.19. Pines del PIC16F87X

3.3.4.2. Microcontrolador ATMEL ATmega32

ATMEL ha desarrollado una gran cantidad de microcontroladores (AVR) en diferentes gamas, de forma similar a lo que ha hecho la empresa Microchip con los

PICs. Una de las diferencias es la velocidad de procesamiento. La arquitectura de los PIC's divide en 4 la frecuencia de oscilación del cristal, y se basa en eso para su funcionamiento, los AVR no, estos trabajan con el reloj puro, sin divisiones, por lo que trabajan 4 veces mas rápido que los PIC's.

Por su facilidad de uso, su bajo coste, y su amplia difusión en el sector de la enseñanza, se ha decidido afrontar el problema del control del sistema haciendo uso de los microcontroladores AVR.

Características principales del ATMEGA32

- Los microcontroladores AVR permiten la ejecución de instrucciones mediante la metodología 'pipeline' con dos etapas (cargar y ejecutar), que les permite ejecutar la mayoría de las instrucciones en un ciclo de reloj, por lo tanto son más veloces.
- Los registros punteros X, Y y Z tienen capacidades de direccionamiento diferentes entre sí.
- Los registros 0 al 15 tienen diferentes capacidades de direccionamiento que los registros 16 al 31 (Figura III.20).
- El Atmega32 contiene 32K bytes de memoria Flash Re-programable para almacenar el programa. Ya que todas las instrucciones de los AVR son de 16 o 32 bits de ancho, la Flash se organiza en 16K x 16. Posee 1024 Bytes de EEPROM y 2K Byte para SRAM interna
- La frecuencia de trabajo para el atmega32 puede variar entre 0 y 16 Mhz
- Bajo consumo a 1 Mhz es de 1.1mA

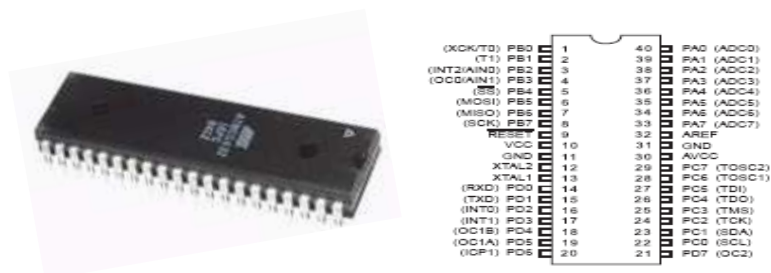


Figura III.20. AVR ATmega32 y su disposición de pines

3.3.5. Métricas para la elección del microcontrolador.

Entre las características que ofrecen los microcontroladores se debe tomar en cuenta para su elección:

- **Sencillez de programación, manejo y grabación:** El encapsulado se recomienda que sea de tipo DIP para facilitar el modo de grabación y realizar las pruebas en un Bread Board.
- **Alta velocidad:** Cuando se miden y controlan variables como son la temperatura y humedad se debe procurar obtener un sistema en tiempo real. Mientras más rápido sea el procesamiento, se obtendrán mejores resultados.
- **Capacidad de Memoria:** El microprocesador debe tener la memoria de programa (Memoria Flash) en un rango entre 16 Kb y 32 Kb, pues debido a que realizará todos los procesos para la conexión con la red, y el alojamiento de la página Web se necesitará un mayor recurso de memoria.
- **Número disponible de puertos:** Se necesita un amplio número de pines de entrada y salida que se analizará en mejor detalle en el siguiente capítulo.
- **Bajo precio:** Esta métrica tiene gran importancia para que permita introducir al mercado el producto con mayor facilidad.

3.3.6. Estudio de la mejor alternativa de microprocesador

Cabe mencionar que los microprocesadores seleccionados tienen soporte del protocolo SPI, debido a que el módulo anteriormente seleccionado (ENC28j60) trabaja con este tipo de comunicación.

De la misma forma aplicamos el Método Ponderado para elegir el microcontrolador. Para lo cual asignamos pesos o valores para la ponderación de 0 a 10 (Tabla III.9).

Tabla III.9. Asignación cuantitativa para las métricas.

Métricas o factores	Valores para la ponderación	Ponderación
Programación, manejo y grabación	9	0,20
Velocidad	8	0,18
Memoria	10	0,22
Puertos disponibles	10	0,22
Bajo precio y disponibilidad	8	0,18
Total	45	1,00

Tabla IV. 10 Precios de los microcontroladores

Métricas o factores	PRECIO \$ (USD)
PIC16f877A	7,50
ATMEGA32	9

FUENTE: ELECTRO SOL – Riobamba – Ecuador

De la misma forma asignamos calificaciones o costos según la información presentada de los microcontroladores (Tabla III.11.). Por ejemplo: en relación a la velocidad el Atmega32 es 4 veces más veloz que el 16F877A por lo tanto la calificación de la velocidad del AVR respecto al PIC seleccionado será 4 veces la calificación o costo que le asignemos al PIC. Se calificó con valores de 1 a 5 indicando Insuficiente, Regular, Bueno, Muy Bueno y sobresaliente respectivamente.

Tabla III.11. Calificaciones y costos

Factor Alternativa	Sencillez, manejo y programación	Velocidad	Memoria	Puertos disponibles	Bajo precio USD (\$)
Atmega32	4	5	5	5	9
16F877A	4	1	1	5	7,50

El procedimiento para obtener el costo del precio es el mismo que se realizó para conseguir el costo del dispositivo Ethernet. El proceso que se debe realizar consta:

- Identificamos el valor menor, en este caso es \$7,50. Ese valor representa la calificación 10.
- Calculamos en exceso del otro valor, es decir, se resta \$9 menos \$7,50. El resultado de esta operación es: 2
- Realizamos la siguiente regla de tres:

$$\text{Costo} = \frac{1,5 * 10}{7,5} = 2$$

Con la información de la tabla III.9 obtenemos la matriz de puntos para determinar cual microcontrolador es el adecuado. El microcontrolador escogido será el que tenga un mayor costo.

Tabla III.12. Matriz de puntos

Factores	Ponderación	Atmega32		16F877A	
		Calif	Calif. P	Calif.	Calif. P
Sencillez, manejo y programación	0,20	4	0,8	4	0,8
Velocidad	0,18	5	0,9	1	0,18
Memoria	0,22	5	1,1	1	0,22
Puertos disponibles	0,22	5	1,1	5	1,1
Bajo precio y disponibilidad	0,18	2	0,36	10	1,8
total	1,00	-	4,26	-	4,1

De la tabla anterior (Tabla III.12) determinamos que el microcontrolador adecuado para el presente proyecto es el ATmega32.

3.3.7. Resultados de la elección

Con la aplicación del Método ponderado se determinó que la mejor alternativa para el sistema Web Embebido a implementar esta integrado por el microcontrolador ATMEGA32 y para dar la conectividad con la red la opción más adecuada fue el ENC28J60.

3.4. Software de Desarrollo

La empresa Atmel básicamente dispone de dos herramientas gratuitas como son el AVR Studio y WinAVR (complemento para AVR Studio) que soporta lenguaje ensamblador y de alto nivel como es C/C++. Otras herramientas de desarrollo soportan lenguaje BASIC como el caso de Bascom, pero es muy poco versátil

3.4.1. AVR STUDIO

El software que se utilizó para el desarrollo del proyecto fue AVR Studio 4, una vez instalado el programa, se puede acceder a través del botón Inicio de Windows.

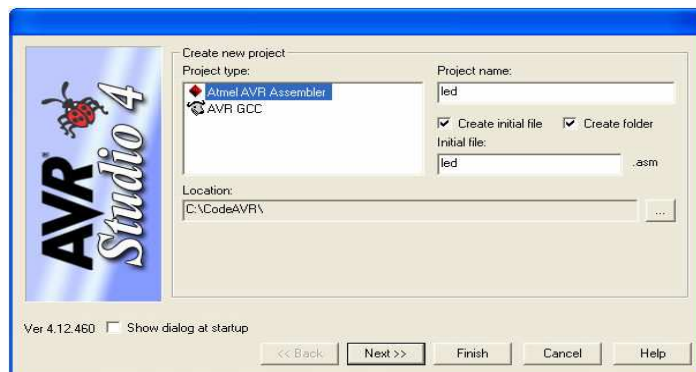


Figura III.21. Pantalla de elección de tipo de proyecto

En este software se pueden realizar dos tipos de proyectos (Figura III.21):

- Atmel AVR Assembler: si se desea programar en ensamblador (.asm)
- AVR GCC: para realizar programas en Lenguaje C (.c)

Se debe seleccionar el lugar donde se guardara el proyecto. Este es el lugar donde AVR Studio almacenara todos los archivos asociados con este proyecto. Si la carpeta no existe, AVR Studio crea automáticamente.

AVR Studio ofrece también la opción de escoger la plataforma para el debugger a utilizar y el dispositivo. Cuando no se dispone de un debugger se selecciona AVR Simulator, para simular los programas en caso de necesitar depurar alguno de ellos. El dispositivo elegido para este proyecto fue el atmega32, se selecciona ese modelo en la ventana de configuración.

Después de dar un clic en "Finish", el administrador de proyectos mostrará una figura como la Figura III.22. En esta vista, se verán todos los archivos asociados con el proyecto

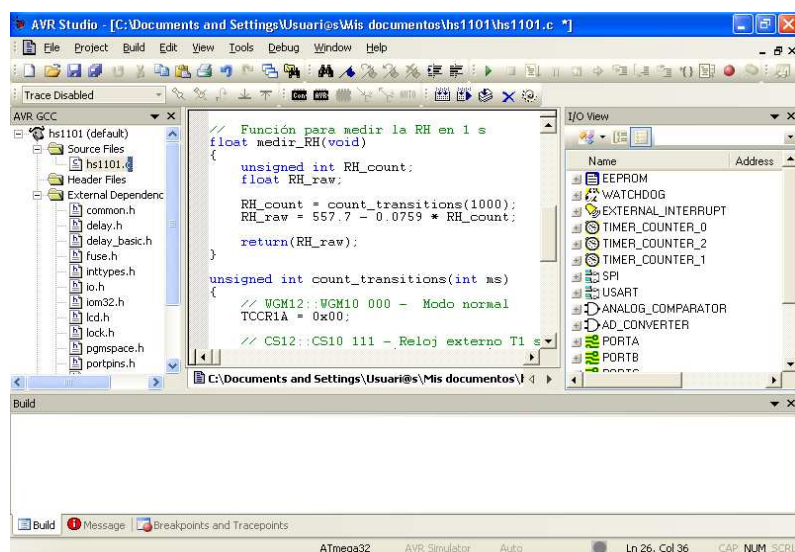


Figura III.22. Pantalla principal del AVR Studio

En caso de duda siempre es bueno mirar el archivo de definiciones correspondiente al microcontrolador que se esté usando. Sobretudo cuando se necesita verificar si un registro o puerto está definido o cuál es su nombre.

Para compilar, es decir, traducir el código escrito en C a un archivo binario/hexadecimal en AVR Studio, se presiona la tecla F7 (que no solo compila, sino que construye el proyecto completo).

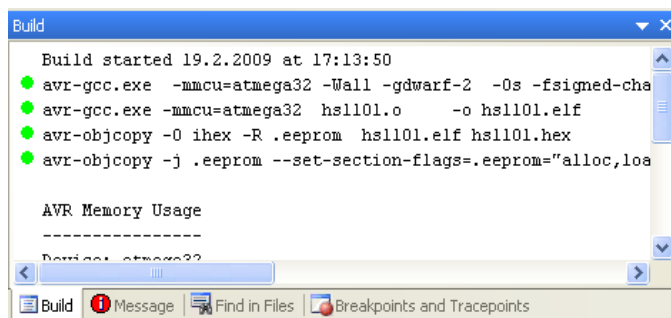


Figura III.23. Área de información de proyecto ejecutado

Si se observa en la Figura III.23, el modelo del microcontrolador se pasa al compilador por medio de la línea de comandos, así como también el nivel de optimización.

Nivel de optimización: Es un ajuste que dependerá de la aplicación en desarrollo, pero en general se utiliza -Os (Figura III.24), que minimiza el tamaño del código. Utilizar optimizaciones que reduzcan el tiempo de ejecución no se recomiendan, pues en muchos casos pueden expandir el código de los ciclos lo que trae una pérdida de espacio en la memoria de programa. Cuando se programa en microcontroladores en general, demorar unos cuantos ciclos más de reloj no es un problema, pero sí lo es que el tamaño del programa sea extenso pues en la mayoría de los casos es ese el recurso limitado.

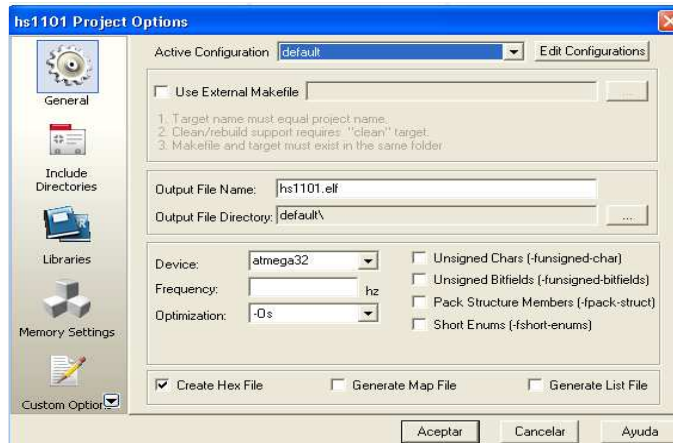


Figura III.24. Opciones de configuración del proyecto

3.4.2. PROGISP

Un programador consta de dos partes, un software para el computador (Figura III.25) y un aparato físico (AVR USBISP) que se encargan de traspasar el archivo hexadecimal, al microcontrolador que debe ejecutar el programa.

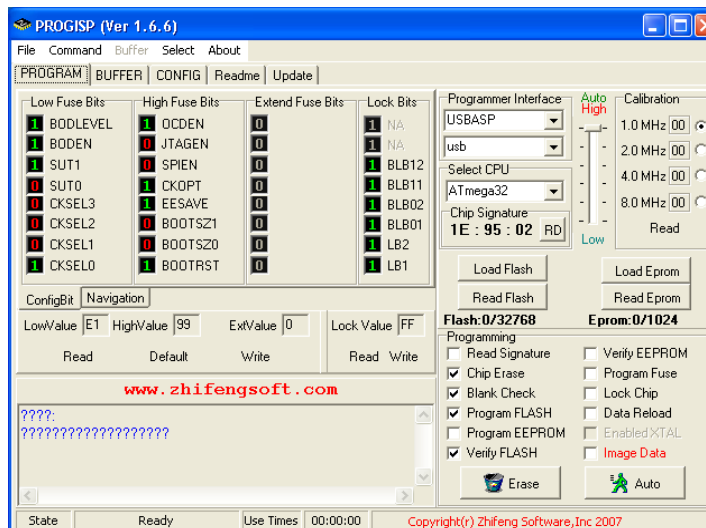


Figura III.25. Software Programador PROGISP

En este software se configura y programa los fuses. Los fuses son bits que determinan la frecuencia a la que el microprocesador trabajará. Los dos bytes de

fuses, están divididos en fuses alto y fuses bajo. A continuación una breve descripción de estos bits:

En el registro de fuses alto se configuran los siguientes:

- OCDEN: Habilita algunos osciladores a pesar de estar en modo sleep.
- JTAGEN: Habilita el JTAG, interfaz que cumple con el estándar 1149.1 de la IEEE.
- SPIEN: Habilita o deshabilita el uso del ISP.
- CKOPT :su funcionalidad depende de los bit de CKSEL
- EESAVE: Indica si se borra la memoria eeprom o no durante el ciclo de borrado.
- BOOTSZ1: Configura el tamaño del arrancador
- BOOTSZ0: Cargador
- BOOTRST: Selecciona donde comienza el vector del reset

En el registro de fusibles bajo se configuran los siguientes:

- BODLEVEL: Indica el nivel en el que se detecta el nivel de bajo voltaje
- BODEN: Habilita el detector de nivel de bajo voltaje
- SUT1, SUT0 : Indica el tiempo que debe de esperar antes iniciar el programa dentro del microcontrolador
- CKSEL3, CKSEL2, CKSEL1, CKSEL0: Se utiliza para seleccionar los tipos de reloj a utilizar desde el oscilador interno de 1 Mhz hasta 8 Mhz internos, o los osciladores externos que alcanzan hasta los 16Mhz.

Para trabajar con la máxima frecuencia para un atmega32 se utilizó un cristal externo de 16Mhz, configurando los fuses alto con un valor de 99 y los fuses bajo con un valor hexadecimal de FF. La programación de los fuses se puede realizar por una sola vez.

CAPITULO IV

DISEÑO Y DESARROLLO DEL SISTEMA WEB EMBEBIDO

4.1. Requerimientos del sistema

En esta etapa se definió lo que hará el sistema que se está implementando. Un sistema que se define sin considerar los requerimientos funcionales y financieros del mercado tiene una probabilidad ínfima de éxito.

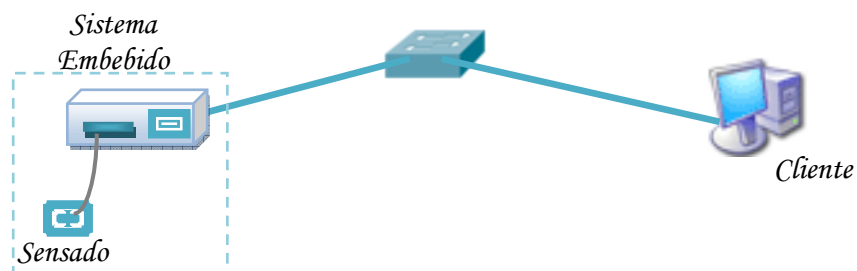


Figura IV.26. Sistema Web Embebido implementado

En este proyecto se requirió diseñar e implementar un Sistema Web que permita controlar la temperatura y humedad (Figura IV.26). Este tipo de circuitos es ideal para la utilización en hogares o, mejor aún, en invernaderos o cámaras de cría de diferentes animales, cuartos de equipos electrónicos, almacenamiento de granos, habitaciones de hospital, en la industria textil etc.

El diagrama de bloques del Sistema se presenta en la Figura IV.27.

Los sensores se encargarán de la adquisición de datos de temperatura y humedad para luego ser procesados por el microcontrolador.

Para realizar el control de las variables o valores medidos existen diferentes técnicas como: ventilación, refrigeración, humidificadores, calefacción, etc.

Algunos ejemplos de alternativas de control incluyen métodos como por ejemplo: para aumentar la humedad relativa se reduce la temperatura; otro método consiste en añadir humedad al aire utilizando nebulizadores, vaporizadores.

En este proyecto se puede seleccionar el encendido y apagado del sistema de ventilación manualmente. También se podrá encender la calefacción según sea el caso.

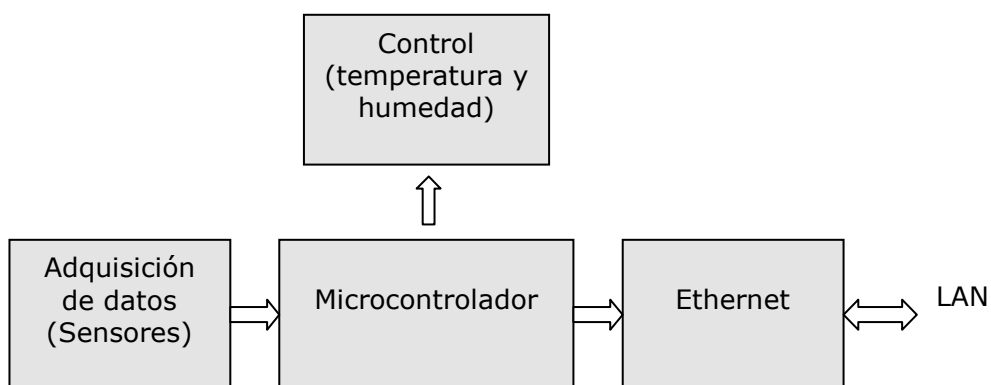


Figura IV.27. Diagrama de bloques del Servidor Web Embebido

Como aplicación este prototipo está dirigido a mantener una temperatura y humedad adecuada en la utilización de equipos electrónicos (Figura IV.27). El rango de temperatura que se determinó fue de 15 °C -21°C mientras que el de humedad está entre 35%-60%, no debe disminuir de 35% porque a menor humedad aumenta la electricidad estática y por lo tanto aumenta el riesgo de un equipo dañado. Si los valores medidos de las variables mencionadas están fuera de ese

rango se activarán dispositivos orientados a mantener la temperatura y humedad ideales.

El prototipo podría ser utilizado también en el área de procesamiento de madera, pues se requiere para la temperatura un rango de 18-21°C y humedad relativa de 35 a 40%, en el almacenamiento de harina, el rango de temperatura es de 18-27°C y de 50-60% de humedad relativa, otra aplicación es el confort humano que esta en una temperatura de 20-22°C y una humedad de 40 a 60%

Sin embargo, el sistema cuenta con un control manual de encendido y apagado del ventilador y calefactor permitiendo la manipulación de estos equipos de acuerdo a la necesidad del usuario.

Para poder acceder a los datos (lecturas de temperatura, humedad y estado ON/OFF del ventilador y calefactor), se puede dirigir a la página web que se encuentra alojada en el dispositivo Web embebido, digitando en un navegador Web la dirección IP del sistema.

Se utilizó direccionamiento estático para asignar una IP al Sistema Web Embebido, pues se recomienda utilizar este tipo de direccionamiento para servidores, interfaces de routers, impresoras, etc.

4.2. Diseño e implementación de hardware

Es importante la simulación del sistema antes de su implementación, porque permite conocer con bastante anticipación el comportamiento del sistema que se esta implementando.

De esta forma se puede corregir en las primeras etapas la funcionalidad.

4.2.1. Adquisición de datos

4.2.1.1 Sensor de humedad

Este circuito es el montaje típico como multivibrador astable diseñado para un 555 (Ver figura VI.28.). Por lo tanto la salida es un tren de pulsos, en Figura VI.29 se muestra la onda de salida de este circuito.

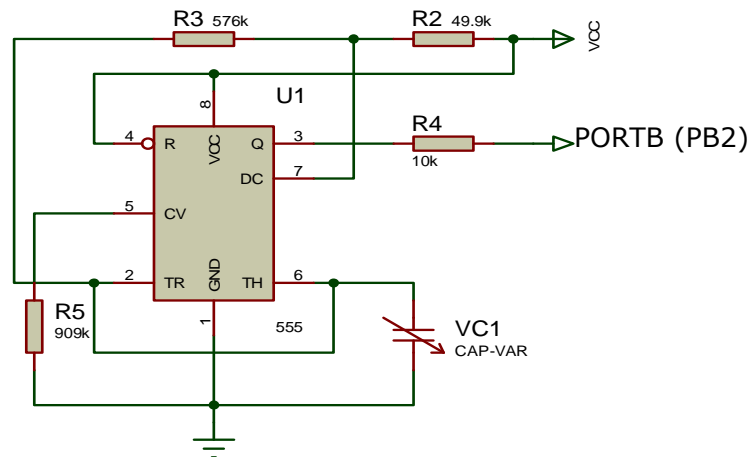


Figura IV.28. Multivibrador astable usando el HS1101

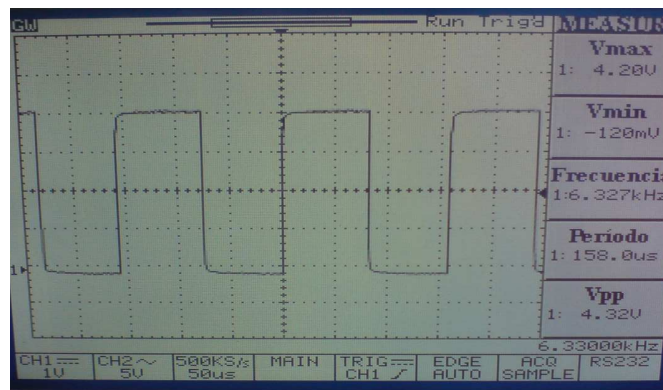


Figura IV.29. Salida del multivibrador astable vista en un osciloscopio

El HS1101, se utiliza como condensador variable, se conecta a las entradas TRIG y THRES y al pin 7 se conecta la resistencia R4. El condensador equivalente formado por el HS1101 se carga a través de R2 y R4 hasta llegar a la tensión umbral (aproximadamente 0,67 Vcc) y se descarga solo a través de R2 hasta llegar a la tensión de Trigger (aproximadamente 0,333 Vcc) ya que en descarga del sensor a

través de las resistencias R2 y R4 determinan el ciclo de trabajo. El 555 debe ser CMOS. La resistencia R3 protege contra cortocircuitos mientras que la resistencia R1 disequilibra la compensación de temperatura interna del 555 e introduce un coeficiente de temperatura emparejado al del HS1101. La compensación de temperatura interior del 555 cambia de un fabricante a otro, el valor de R1 debe adaptarse al chip específico.

La respuesta de este circuito, dada por el fabricante, se resume en la siguiente tabla:

Tabla IV.13. Punto de Referencia a 6660Hz para 55% RH @ 25°C

RH%	0	10	20	30	40	50	60	70	80	90	100
F. Hz.	7351	7224	7100	6976	6853	6728	6600	6468	6330	6186	6033

De ahí se puede deducir la siguiente ecuación, que es una aproximación lineal:

$$RH(\%) = 560 - 0.0758 * F(\text{Hz})$$

En la tabla IV.14. Se calcularon los valores de la humedad según la frecuencia:

Tabla IV.14. Valores de la frecuencia cuando RH = 0% hasta RH = 99%

RH (%)	F (T/C1)	RH (%)	F (T/C1)	RH (%)	F (T/C1)	RH (%)	F (T/C1)	RH (%)	F (T/C1)
0	7388	20	7124	40	6860	60	6596	80	6332
1	7375	21	7071	41	6808	61	6544	81	6281
2	7361	22	7058	42	6794	62	6531	82	6267
3	7348	23	7045	43	6781	63	6518	83	6254
4	7335	24	7032	44	6768	64	6505	84	6241
5	7322	25	7018	45	6755	65	6491	85	6228
6	7309	26	7005	46	6742	66	6478	86	6215
7	7296	27	6992	47	6729	67	6465	87	6202
8	7282	28	6979	48	6715	68	6452	88	6188
9	7269	29	6966	49	6702	69	6439	89	6175
10	7256	30	6953	50	6689	70	6426	90	6162
11	7243	31	6939	51	6676	71	6412	91	6149
12	7230	32	6926	52	6663	72	6399	92	6136
13	7216	33	6913	53	6650	73	6386	93	6123
14	7203	34	6900	54	6636	74	6373	94	6109
15	7190	35	6887	55	6623	75	6360	95	6096
16	7177	36	6874	56	6610	76	6347	96	6083
17	7164	37	6860	57	6597	77	6333	97	6070
18	7150	38	6847	58	6584	78	6320	98	6057
19	7137	39	6834	59	6570	79	6307	99	6043

4.2.1.2. Sensor de temperatura

La ecuación que domina el cambio de resistencia de un termistor respecto a la temperatura está dada por:

$$R_T = R_0 e^{b\left(\frac{1}{T} - \frac{1}{T_0}\right)}$$

Donde:

R_T = Resistencia del termistor.

R_0 = Resistencia inicial, de acuerdo a la hoja de datos del termistor es 10kOhm.

e = Coeficiente del número neperiano.

b = Coeficiente térmico que será reemplazado por 4050.

T_t = Temperatura de trabajo en grados Kelvin.

T_0 = Temperatura de referencia en grados Kelvin, es decir $T_0 = 25 + 273.15$.

La interfaz para el termistor hacia el circuito para medir magnitudes con variación lenta (como es aquí el caso de la temperatura), se puede basar en un puente o en un oscilador. Es decir, en operaciones con tensiones y corrientes o en operaciones con frecuencias, se optó por la primera solución (Figura IV.30).

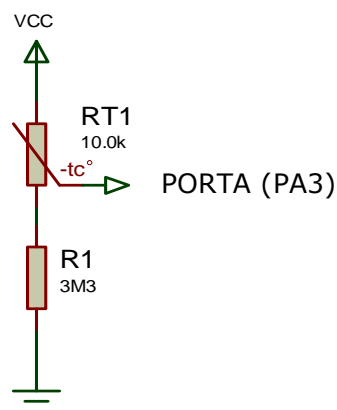


Figura IV.30. Circuito puente utilizado como interfaz al microprocesador

Donde V_{REF} es el valor proporcionado por el microprocesador en este caso el ATmega32, R_T es el valor de la resistencia del termistor a un valor de temperatura

dato calculada mediante la formula anterior y el valor de R es 10 kΩ, dicho valor es el recomendado en su respectiva hoja de datos. El voltaje entregado en VOUT dependerá de la variación de la resistencia RT, la misma que depende de la variación de la temperatura, y para calcularla se utiliza la formula

$$V_{\text{ant}} = 2.56 \times \frac{10k}{10k + R_T}$$

En vista que el ATmega32 posee un modulo ADC con 10 bits de resolución, $2^{10} = 1024$, la codificación digital de la entrada analógica a la salida del modulo será dada por:

$$\text{ADC} = 1024 \times \frac{V_{\text{OUT}}}{2.56}$$

Siendo 2.56 el valor del voltaje de Referencia del ATmega32 (Tabla IV.15). Todo el proceso descrito anteriormente fue implementado en el firmware del sistema

Tabla IV.15. Valores ADC desde Tt = 0 hasta Tt = 99

Tt (°C)	ADC	Tt (°C)	ADC	Tt (°C)	ADC	Tt (°C)	ADC	Tt (°C)	ADC
0	229	20	453	40	673	60	826	80	914
1	239	21	465	41	682	61	832	81	917
2	249	22	477	42	692	62	837	82	920
3	259	23	489	43	701	63	843	83	923
4	270	24	500	44	710	64	848	84	926
5	280	25	512	45	718	65	853	85	929
6	291	26	524	46	727	66	858	86	931
7	302	27	535	47	735	67	863	87	934
8	313	28	547	48	743	68	867	88	936
9	324	29	558	49	751	69	872	89	939
10	335	30	569	50	759	70	876	90	941
11	347	31	580	51	766	71	881	91	944
12	358	32	591	52	774	72	885	92	946
13	370	33	602	53	781	73	889	93	948
14	382	34	613	54	788	74	893	94	950
15	394	35	623	55	795	75	897	95	952
16	405	36	633	56	801	76	900	96	954
17	417	37	644	57	808	77	904	97	956
18	429	38	654	58	814	78	907	98	958
19	441	39	663	59	820	79	911	99	960

4.2.2. Microcontrolador

Los AVR utilizan la Arquitectura Harvard, con el bus de datos y el bus de memorias separados. Los 32 registros de propósito general de 8 bits se localizan al inicio de las direcciones de memoria de la SRAM. A continuación se presenta el diagrama de bloques del ATmega32 en la Figura IV.31

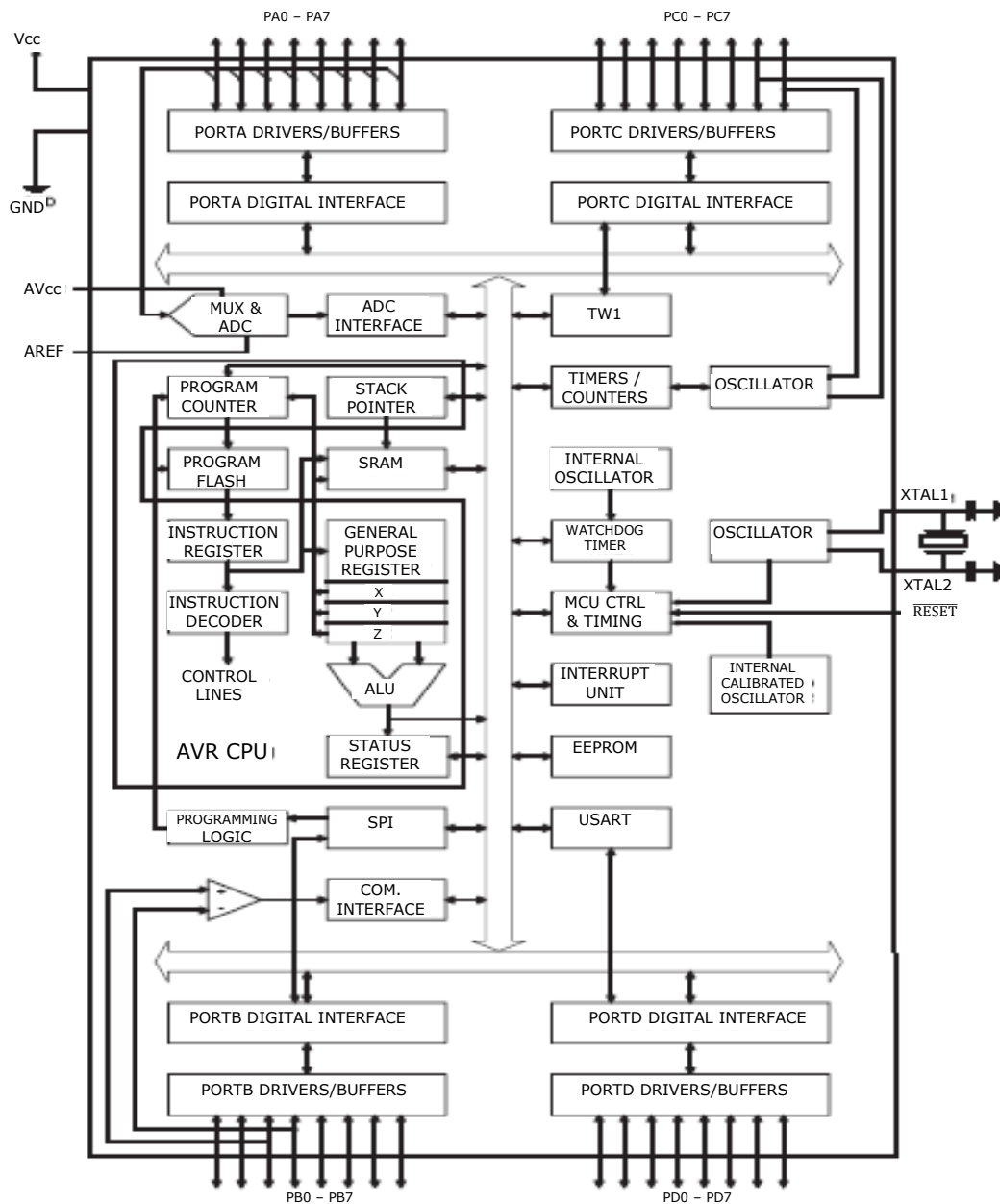


Figura IV.31. Diagrama de bloques interno del atmega32

Una parte principal del AVR es el núcleo del CPU del AVR, pues su función es asegurar la correcta ejecución de un programa. El CPU debe ser capaz de acceder a la memoria, ejecutar cálculos, controlar los periféricos y manejar las interrupciones. El Microprocesador es la parte central del hardware (Figura IV.32) y este bloque es implementado en su totalidad con el firmware del Sistema.

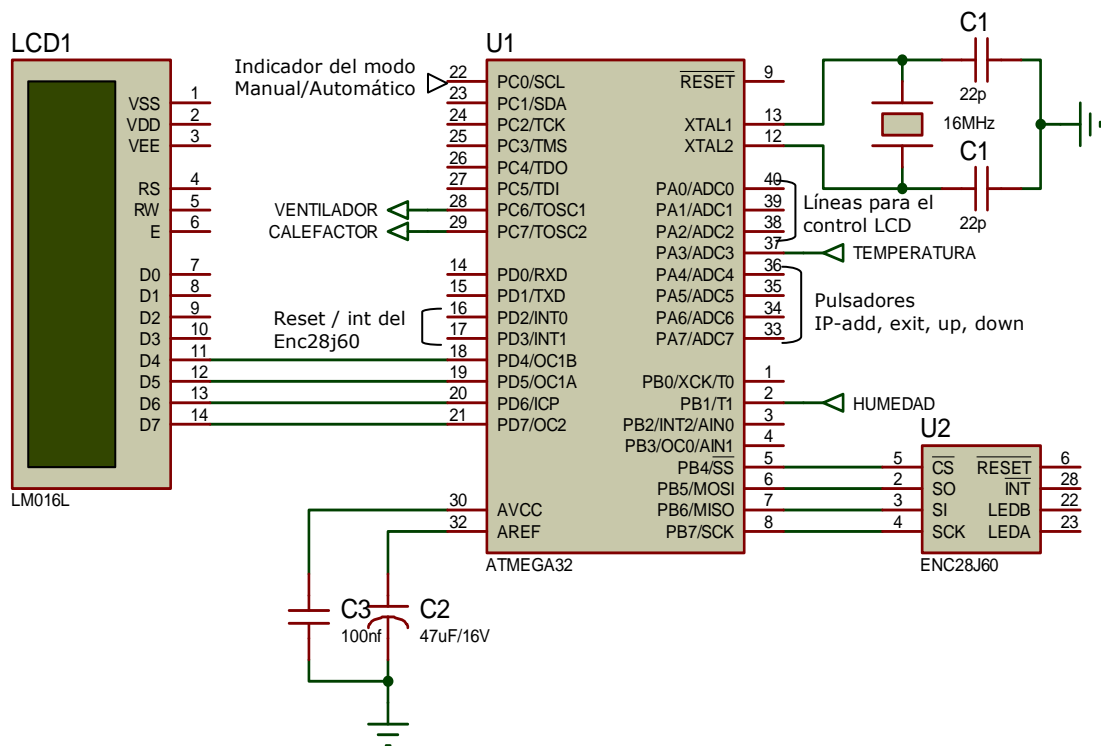


Figura IV.32. Diagrama de hardware para el microprocesador

La frecuencia del sistema es de 16 Mhz, el motivo para trabajar con esta frecuencia se explicará más adelante. El número total de pines utilizados fue de 26 de 32 disponibles.

4.2.3. Control del sistema

Este bloque esta formado por tres partes: Configuración a través de pulsadores de la dirección IP del servidor, la visualización de la temperatura, humedad y

configuración de la dirección mediante la pantalla LCD, y dos interfaces de potencia para el ventilador y calefactor respectivamente.

4.3.2.1. Control dirección IP

Para que el dispositivo sea más amigable al usuario y más competitivo con las alternativas existentes en sistemas embebidos se diseñó el control para configurar el ingreso de la dirección IP del servidor.

El control para la configuración local se diseño con cuatro pulsadores (Figura IV.33) a los cuales se los llamo IP_Add, EXIT, UP y DOWN.

- **IP_add:** Al pulsar este botón se ingresa al modo de configuración de dirección IP.
- **EXIT:** Al pulsar este botón permitirá salir de la Configuración de la dirección.
- **UP:** Permite el desplazamiento entre la visualización de la dirección IP configurada y las mediciones de temperatura y humedad. Además permite mover cada octeto de una dirección IP en dirección superior.
- **DOWN:** Este botón realiza las mismas funciones que el anterior pero en dirección contraria

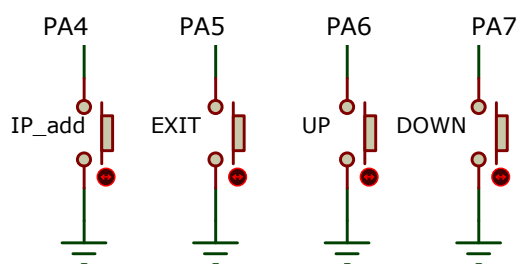


Figura IV.33. Control para la configuración de la IP

Las interfaces de los pulsadores no se manejaron con resistencias PULL-UP como generalmente se utiliza al trabajar con microprocesadores puesto que el AVR tiene

conectado este tipo de resistencias internamente, por lo tanto solo se necesita configurar mediante el Firmware el PULL-UP para el puerto correspondiente.

4.3.2.2. Visualización

La configuración mediante el control de mando fuera imposible sin un método de visualización de lo que se está configurando, además se necesita un medio de visualización local del estado de las mediciones del sistema, todo esto se hizo mediante una pantalla LCD.

La pantalla LCD que se utilizó es del tipo *alfanumérico*, específicamente el modelo de 2 líneas y 16 caracteres por línea. En la hoja de datos del módulo LCD, el fabricante informa el diagrama de tiempos y estados de las señales, para establecer la comunicación con el módulo LCD, tanto para enviar ordenes de configuración o datos a escribir en el display. En la tabla IV.16 se realiza una breve descripción de los pines del display.

Tabla IV.16. Descripción de pines del LCD 2X16

Pin	Simb.	Descripción
1	Vss	Tierra de alimentación GND
2	Vdd	Alimentación + 5V CC
3	Vo	Ajuste de contraste del cristal liquido (0 a +5)
4	RS	Selección de datos control/datos RS=0 reg. Control RS=1 reg. Datos
5	R/W	Lectura/escritura en LCD R/W=0 escritura R/W=1 lectura
6	E	Habilitación E=0 modulo reconocido E=1 modulo conectado
7	D0	D0 – D7 reciben los caracteres ASCII a representar, así como ciertos códigos de control que regulan los efectos de visualización
8	D1	
9	D2	
10	D3	
11	D4	
12	D5	
13	D6	
14	D7	Bit mas significativo (bus de datos bidireccional)
15	A	Alimentación del backlight =3,5V o =5V CC
16	K	Tierra GND del backlight

El LCD dispone de BACKLIGHT (Retroiluminador) el cual ayuda a visualizar la información en lugares donde la luz natural es escasa.

Las conexiones para controlar la visualización de datos sobre el LCD se hicieron usando una interfaz de 4 bits, pues con esto se ahorra pines de salida del microprocesador al usar únicamente 4 de ellas en lugar de las 8 que requeriría usar en una interfaz de 8 bits.

Se escogió el puerto A del ATMEGA32 para realizar las conexiones de control con el LCD y el puerto D para el envío y recepción de datos hacia el LCD (Figura IV.34).

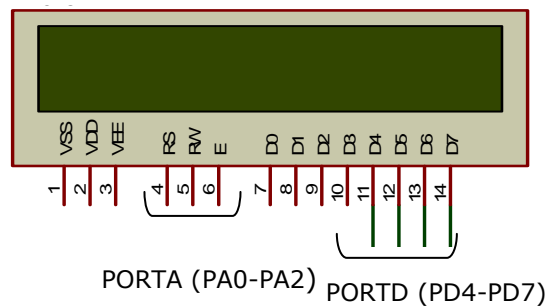


Figura IV.34. Conexiones físicas del LCD

4.3.2.3. Interfaz de potencia

Para activar o desactivar los actuadores (ventilador y calefactor) empleados para controlar la temperatura y humedad se utilizaron los pines 29 y 28 del AVR que pertenecen al PUERTO C (PC6 y PC7 respectivamente).

Para el diseño de la interfaz de potencia se utilizó un octoacoplador y un triac (Figura IV.35). La señal digital proveniente del AVR es aplicada al cátodo del LED interno del optoacoplador. El ánodo de ese diodo es cableado a MASA digital. El brillo producido por el LED acciona el Triac del opto, que, a su vez, acciona el triac de potencia. La red RC conectada en paralelo con el triac de potencia limita la velocidad de evolución de la tensión ante cargas inductivas.

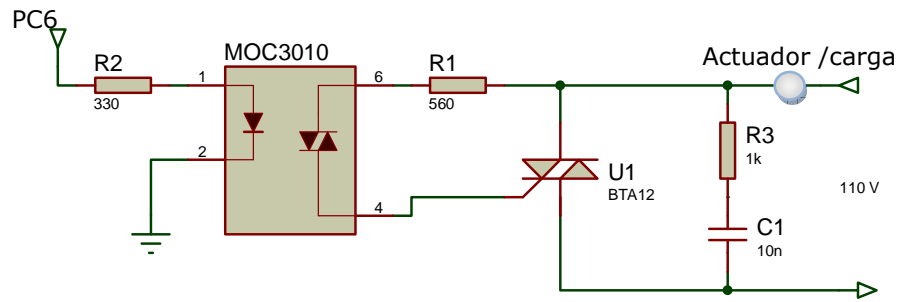


Figura IV.35. Interfaz de Potencia a través de un Octoacoplador

Es indispensable montar el Triac en un buen disipador térmico. El circuito mostrado en la Figura IV.35 corresponde a un solo canal. Para montar un sistema de mas canales repetir este tantas veces como canales necesarios. Se debe conectar sólo un circuito a cada pin del puerto del AVR.

4.3.3. Diseño de la interfaz con la red

El microcontrolador seleccionado para esta aplicación el ATMEGA32 y el modulo ENC28j60 manejan el protocolo SPI (Ver figura IV.36.). El módulo de Ethernet reúne todas las especificaciones de IEEE 802.3 para conectividad de una red.

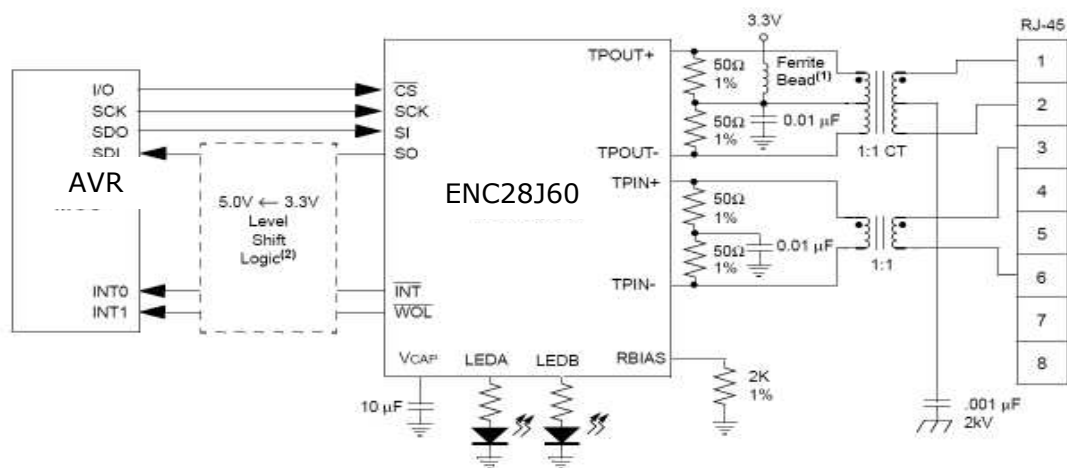


Figura IV.36. Conexión entre el ATMEGA32 y ENC28J60

Las conexiones con el ENC28J60 se realizaron como especifica el fabricante, MICROCHIP (2). Se utilizó los pines designados para SPI del microprocesador en el puerto B. Hay que tomar una consideración adicional, debido a que la pantalla LCD solo funciona con 5V, el microprocesador tiene que funcionar también con 5V, sin embargo el ENC28J60 solo funciona a 3.3V, lo cual es un problema por lo que se utilizó un regulador de voltaje LM394 para obtener los 3.3V.

El ENC28J60 solo funciona a 3.3V, las salidas entregadas también tendrán ese nivel de voltaje. Entonces se utilizaron Buffers Tri-State (74AHC125) para realizar un cambio de nivel de 3.3V a 5V que acepta el microprocesador. Todo esto se ve en más detalle en la Figura IV.37

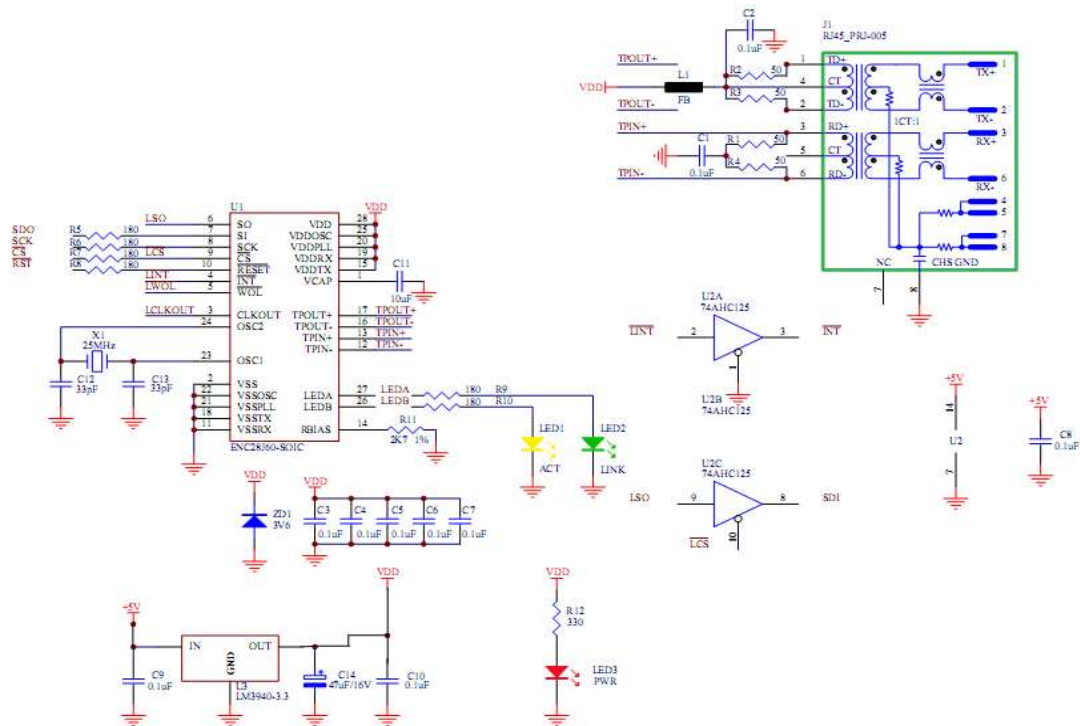


Figura IV.37. Diagrama de conexiones del ENC28J60

4.3. Diseño e implementación de software

Se debe elegir antes un software base para programar el microcontrolador puede ser: lenguaje Ensamblador o de alto nivel, un compilador y un software

programador que junto a su hardware permita bajar el programa ejecutable (con extensión .HEX) a la memoria Flash del micro. El programa que maneja al microcontrolador está desarrollado por módulos, aprovechando una de las ventajas que proporciona el lenguaje C.

Los archivos principales que conforman el firmware del sistema son:

- **main.c:** Archivo principal del sistema contiene las estructuras centrales en las que se basa el sistema.
- **ad.c:** Contiene todas las funciones que permiten la adquisición de datos.
- **menu.c:** Contiene las funciones principales que permiten el control del sistema local.
- **lcd.c:** Contiene todas las funciones que permiten el control de la pantalla LCD para visualizar la información
- **enc28j60.c:** Contiene todas las funciones que permiten el manejo de la interfaz hacia Ethernet mediante el circuito integrado ENC28J60.
- **arp.c:** Contiene las funciones que permiten la gestión del protocolo ARP.
- **icmp.c:** Contiene las funciones que permiten la gestión del protocolo ICMP.
- **tcp.c:** Contiene todas las funciones que permiten el funcionamiento del protocolo TCP.
- **http.c:** Contiene todas las funciones que implementan el protocolo HTTP como la generación de la página Web

4.3.1. Programa principal del sistema

El programa principal del sistema se basó en el diagrama mostrado en la Figura IV.38, el código fuente se encuentra definido en el archivo main.c, en este archivo se define lo siguiente:

- La inicialización de variables locales y del sistema, es decir las variables que almacenarán la dirección IP y la dirección MAC del servidor para su comunicación con la red.

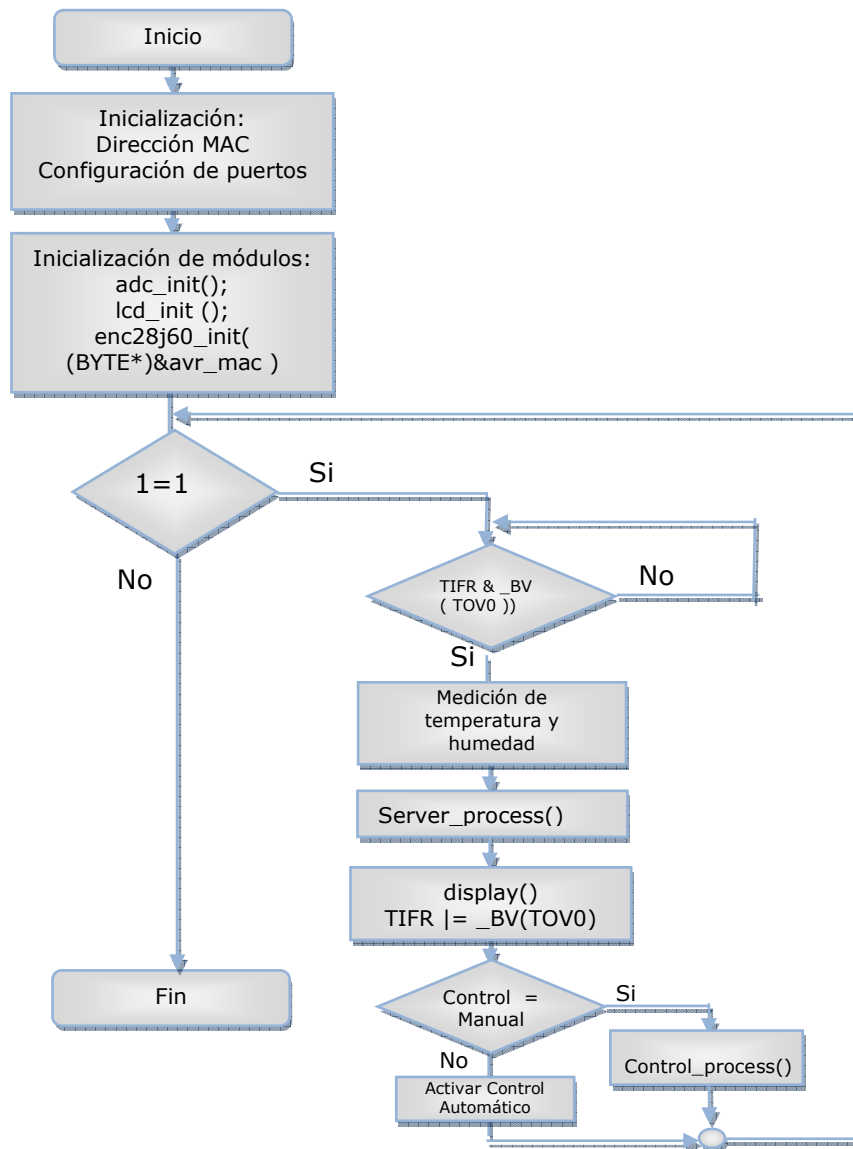


Figura IV.38. Diagrama de flujo principal del sistema

- En la programación de esta familia de microprocesadores es recomendable la asignación de valores mediante el intercambio bit a bit del registro, mas no una asignación directa, por ello es muy común usar la macro `_BV()` que permite un desplazamiento hacia la izquierda bit a bit del valor `0x01`, con esto se selecciona la posición del bit que será puesto a "1" dentro de un registro y asignarle el valor con operaciones OR o NAND.
- Se inicializa el módulo ADC del AVR y el lcd para la visualización de datos.

- Desde el programa se invoca a los procedimientos para la adquisición de datos y el proceso Servidor que esta vinculado a todas las funciones que permiten la conexión con la red.
- Se habilitó la configuración PULL-UP del microprocesador en cada puerto que recibe señal de entrada. Cada pin del puerto consiste de tres registros de bits: DDRxn, PORTxn y PINxn. El bit DDxn en el Registro DDRx selecciona la dirección de ese pin. Si se le escribe a PORTxn un uno lógico cuando el pin esta configurado como un pin de entrada, el resistor pull-up se activa. Para desactivar al resistor de pull-up, se le escribirá un cero lógico a PORTxn o se configura el pin como salida. Los pines de los puertos son de tres-estados cuando se tiene la condición de reset. Las configuraciones de los Puertos (Tabla IV.17) de los pines se muestran a continuación:

Tabla IV.17. Configuraciones de los Puertos

DDxn	PORTxn	I/O	Descripción
0	0	Input	El resistor pull-up se desactiva
0	1	Input	El resistor pull-up se activa
1	0	Output	No
1	1	Output	No

4.3.2. Proceso para el controlador Ethernet

El archivo con todas las funciones que están involucradas con la comunicación con la red están en el archivo enc28j60.c, las funciones principales en este archivo son:

enc28j60_init

- Inicializa la comunicación SPI del ATmega32 asignando 0x50 y 0x01 a los registros *SPCR* y *SPSR*, para que el AVR trabaje en modo MASTER y el ENC28J60 en modo SLAVE, con una frecuencia de reloj de 8 Mhz.

- Se envía a un comando Soft Reset para reiniciar el ENC28J60.
- Definición del tamaño del Buffer para transmisión y recepción de datos. El Buffer total es de 8 kb, se dividió un espacio desde el bit 0 hasta el 6691 para el buffer de recepción de datos y desde el bit 6662 hasta el 8192 (1500 bits, máximo tamaño de la trama ethernet) para el buffer de transmisión.

enc28j60ReadOp y enc28j60WriteOp

- Para el envío/recepción de datos sobre el ENC28J60, en estas funciones se utilizan llamadas a las MACROS como CSACTIVE, para empezar la transmisión SPI al poner un "0" en la línea CS y CSPASSIVE, para terminar la comunicación SPI al poner un "1" en la línea CS.
- El ENC28J60 posee registros de control, repartidos en cuatro bancos diferentes. La función **enc28j60SetBank** se encarga de conmutar a través de esos bancos, y por ende las funciones **enc28j60Write** y **enc28j60Read** hacen uso de las funciones anteriormente descritas para realizar un control total de lectura/escritura sobre el ENC28j60, en otras palabras se utilizó estas dos funciones para realizar toda la gestión de comunicación SPI. El ENC28J60 posee adicionalmente registros de control físicos encargados de toda la gestión de capa física del modelo OSI y para realizar operaciones de lectura/escritura de estos registros se utilizaron las funciones **enc28j60_read_phyreg** y **enc28j60PhyWrite**.

enc28j60_packet_send y enc28j60_packet_receive

- Para completar la comunicación con la interfaz Ethernet son necesarias las rutinas para leer y escribir sobre la Red misma.

4.3.3. Firmware para medición de temperatura y humedad

Para el procesamiento de los datos obtenidos de los sensores de temperatura y humedad se implementó el archivo **ad.c**, el cual tiene rutinas para obtener la temperatura inicializando el modulo ADC interno del ATMEGA32 para luego realizar

la conversión analógica a digital. Las funciones que se utilizan para la adquisición de la temperatura son:

adc_init

- Configuración del registro *ADMUX* que requerirá la referencia de voltaje se va a utilizar, forma de retorno de resultados y canal del ATmega32 que se usará.
- Inicializa el módulo ADC del ATmega32 mediante el registro *ADCSRA*, y configurar el divisor de pre-escala, este último es el factor de división entre la frecuencia del cristal piezoeléctrico y la entrada del reloj al módulo ADC, este factor será de 128 lo que significa que:

$$Reloj_{ADC} = \frac{16 \text{ Mhz}}{128} = 125 \text{ Khz}$$

adc_read

- El inicio de la conversión se realiza mediante el cambio de estado a "1" en el bit *ADSC* del registro *ADCSR*.
- Finalmente se retorna el valor de la conversión ADC mediante la lectura de los registros *ADCL* y *ADCH*, estos registros tienen una longitud de 16 bits.

Para obtener la humedad relativa no se necesita de un ADC como en el caso de la medición de temperatura porque el circuito que se implementó entrega una salida digitalizada. Como se explicó en el capítulo xxxx la frecuencia entregada por el circuito varía de acuerdo a la humedad, por lo tanto se configuró el modo *TIMER/COUNTER1* del AVR (Anexo 3) para contar los pulsos ingresados en 1s a través del Pin 1 del puerto B, es decir la entrada T1. Las funciones que permiten obtener el valor de la humedad son:

count_transitions()

- Configuración del registro *TCC1RA* en modo normal, sólo eventos de sobre flujo, para que comience a contar en modo ascendente

- Configuración del registro TCCR1B para seleccionar la fuente externa y el método considerado para comenzar el conteo, es decir, en flanco ascendente o descendente. Para elegir la entrada T1 se pone a "1" los bits CS10 ... CS12.
- El valor del contador se almacena en el registro de 16 bits TCNT1

measure_RH()

- Es la función que devuelve el valor real de la humedad. Luego de obtener la frecuencia en el registro TCNT1, este resultado se compara con una lista de valores almacenada previamente en la memoria del programa.

4.3.4. Proceso servidor

Una comunicación a través de la red se realiza mediante protocolos como los descritos en el Capítulo II. Sin embargo, para la implementación de la comunicación se usaron varios protocolos definidos a lo largo modelo TCP/IP, pues este modelo es más práctico y resume las mismas funciones realizadas por el modelo OSI.

La implementación de todos estos protocolos se realizó a través de la función **server_process()**, el cual sigue el esquema general de la figura IV.39.

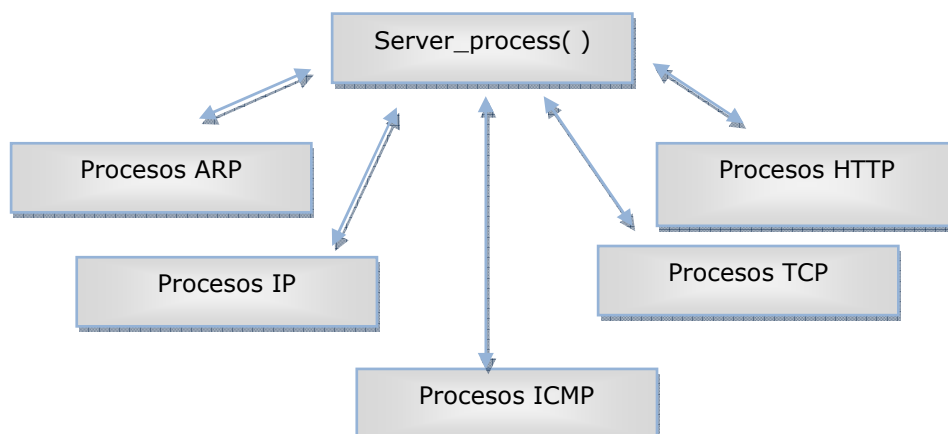


Figura IV.39. Esquema general de la función Server_process()

4.3.3.1. Implementación del protocolo ARP

El archivo que contienen todas las funciones para la implementación del protocolo ARP es **arp.c**, dentro del cual se encuentran funciones como:

arp_who_is

- Implementa las rutinas necesarias para realizar un ARP REQUEST (Petición ARP) y ARP REPLY (Respuesta ARP).
- Para generar un paquete ARP se genera una trama ethernet cuyo código en el campo TYPE es 0x0806 que corresponde al protocolo ARP. Entonces la respuesta del servidor deberá tener el mismo valor en este campo

arp_send_request y arp_send_reply

La generación del paquete ARP va después de la cabecera ethernet y es generada por la función `arp_send_request` y `arp_send_reply`, las cuales ponen los códigos exactos en el paquete ARP. Los valores necesarios para formar la trama se exponen en la tabla IV.18 de acuerdo con la cabecera ARP descrita en el Capítulo II.

Tabla IV.18. Valores necesarios para generar el mensaje ARP

Nombre del Campo	Petición ARP	Respuesta ARP
Tipo (Trama Ethernet)	0x0806	0x0806
Hardware Type	0x0001	0x0001
Protocol Type	0x0800	0x0800
Hlen	0x06	0x06
Plen	0x04	0x04
Operación	0x0001	0x0002
MAC de Origen	local_mac	Dir. MAC del Servidor
IP de Origen	Local_ip	server_ip
MAC de Destino	0x000000000000	local_mac
IP de Destino	server_ip	local_ip

4.3.3.2. Implementación del protocolo IP

Los parámetros para construir la cabecera IP se describen en la tabla IV.19, puesto que el protocolo IP solo implementa un método de identificación en la red.

Tabla IV.19. Valores utilizados en la cabecera IP

Nombre del Campo	Valor asignado
Versión y Longitud de la cabecera IP	0x45 (Versión = 4, Longitud de la cabecera = 5)
Tipo de Servicio	0x00 (Valor por Defecto)
Longitud de la Cabecera IP + el segmento TCP	Este valor se establece en el momento que se tiene el segmento TCP listo
Identificación	0x01 (Valor que identifica al datagrama)
Flags y Offset	0x00
Tiempo de Vida	128 (Valor por defecto)
Protocolo	Este valor se establece en el momento que se tiene el segmento TCP listo
Checksum	Este valor se establece en el momento que se tiene el segmento TCP listo
IP de origen	local_ip
IP de Destino	server_ip

La fragmentación no se contempló en la implementación IP puesto que para el envío de datos planos no es necesario un datagrama mayor a los 1500 bytes, por ende el valor de las banderas (Flags) junto con el desplazamiento (Offset) será 0x00, y el campo identificación será un valor cualquiera. La longitud de la cabecera IP se expresa en grupos de 32 bits (4 bytes), en vista que la cabecera tiene 20 bytes de longitud ($20 / 4 = 5$), 5 será el valor a asignarse en este campo. La rutina principal para la correcta comunicación a nivel de red es:

ip_generate_header

- Se encarga de generar la cabecera del datagrama y los valores adecuados en cada campo del mismo.
- Manipula el datagrama para que se comunique con el servidor

4.3.3.3. Implementación del protocolo ICMP

La implementación de ICMP de basa en la creación del paquete con la asignación adecuada de los valores a los diferentes parámetros que posee este protocolo (tabla IV.20).

Tabla IV.20. Parámetros para una petición ICMP Echo

Nombre del Campo	Valor Asignado
Tipo	8 (Petición Echo)
Código	0 (No tiene significancia en una petición)
Checksum	Este valor se establece al finalizar la creación del paquete.
Identificador	0x10 (Cualquier valor inicializado por el que envía la petición)
Número de Secuencia	0x10 (Cualquier valor inicializado por el que envía la petición)
Datos	Datos al azar hasta completar 32 bytes

Una de las funciones principales es el **icmp_generate_packet** que se encarga de generar el checksum para la cabecera ICMP y si se requiere se añade datos al azar al campo datos con una longitud máxima de 32 bytes (máximo valor permitido), ya que en el caso particular de una petición ICMP Echo se requiere de ello.

La función **icmp_send_request** se implementó para realizar las pruebas de conectividad con el servidor. En esta rutina se creó la petición ICMP Echo

4.3.3.4. Implementación del protocolo TCP

En el archivo **tcp.c** se encuentra principalmente implementada la comunicación en tres vías (Ver capítulo II). La generación del segmento TCP se la realizó mediante la función **tcp_send_packet**.

Para el envío de un segmento a la red primero se genera la cabecera Ethernet, después se pregunta por cualquiera de las dos condiciones: el saludo de 3 vías o la respuesta a una petición anterior.

La tabla IV.21 muestra los valores que se establecieron en el segmento de Inicio de la conexión y finalización de la negociación. Además de los valores que deben esperarse por parte del servidor. El puerto de origen siempre será 1200, pues es un valor cualquiera al azar mayor a 1024.

Tabla IV.21. Valores asignados para el saludo de 3 vías

Nombre del Campo	Saludo de 3 Vías (Valores asignados)		
	Inicio de la conexión	Respuesta del servidor	Finalización de la negociación
Puerto de Origen	1200	80	1200
Puerto de Destino	80	1200	80
Número de Secuencia	0x00000000	0x00000000	0x00000001
Número de ACK	0x00000000	0x00000001	0x00000001
Longitud de la cabecera	0x60	0x60	0x50
Flags	0x02	0x12	0x10
Ventana	0x05ba	El tamaño del buffer que tenga disponible el servidor	0x5ba
Checksum	Se asigna al finalizar todo	El valor que el servidor envíe	Se asigna al finalizar todo
Urgent Pointer	0x0000	0x0000	0x0000
Opciones	0x02040580	0x02040580	No se aplican
Datos	Ningún Dato	Ningún Dato	Ningún Dato

Las funciones **tcp_puts_data** y **tcp_puts_data_p** se utilizaron para asignar datos al segmento. La diferencia es que la primera colocará datos que se encuentran en la memoria RAM del microprocesador, y la segunda asignará datos que están almacenadas en la memoria del programa del microprocesador. Pues casi toda la petición HTTP se almacenó en la memoria del programa y no en la memoria RAM.

4.3.3.5. Implementación del protocolo HTTP

El código empleado para la implementación de este protocolo se encuentra en el archivo **http.c**. El diagrama de flujo se presenta a continuación en la Figura IV.40.

El sistema embebido está configurado para tener un proceso de escucha. Si el paquete receptado por el sistema tiene como puerto destino un servicio web, es decir el puerto 80, se enviará inmediatamente la página web almacenada en la memoria del AVR (la función que se implementa para este fin es **http_home**) con los valores de temperatura y humedad que se obtuvieron de los sensores, además con el estado de encendido o apagado de los actuadores (ventilador y calefactor).

La página se actualizará automáticamente cada 6 s, para evitar la generación de tráfico excesivo en la red. La actualización se logra a través de un JavaScript que puede ser ubicado en cualquier parte la página Web:

```
<script LANGUAGE=JavaScript>
window.setTimeout("window.location.reload()",6000);
</script>
```

El correspondiente código C para el Sistema Web embebido sería:

```
dlen=tcp_puts_data_p (rxtx_buffer,PSTR("<SCRIPT LANGUAGE=JavaScript>
window.setTimeout(\"window.location.reload()\",6000);</SCRIPT>\n"),
dlen);
```

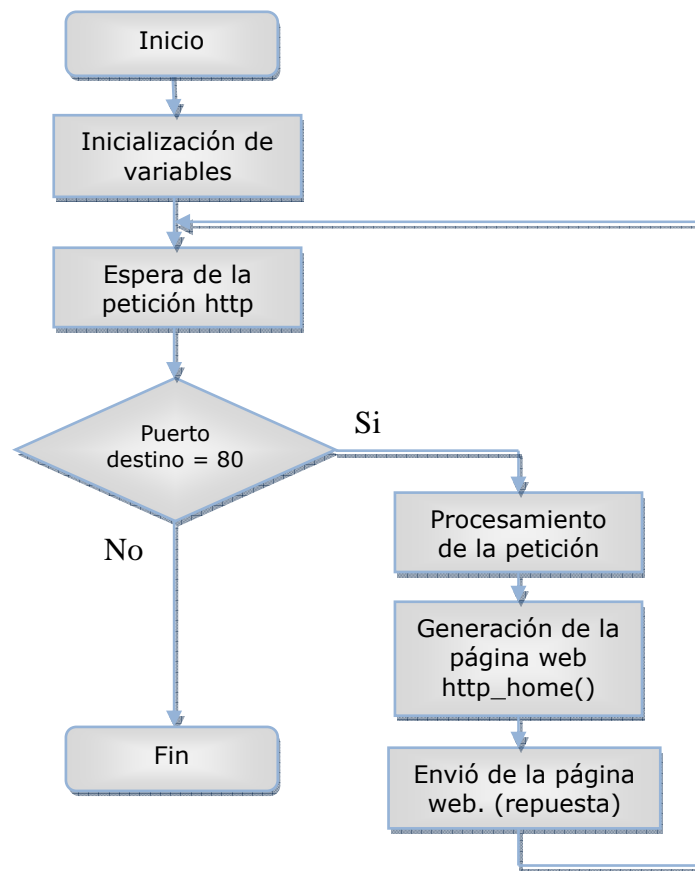


Figura IV.40. Diagrama de Flujo del protocolo HTTP

La función que se encarga de procesar la petición es **http_webserver_process**, que establece la conexión mediante la comunicación en tres vías de TCP antes de enviar la información.

Dentro de esta función se implementa otra, **http_get_variable** que procesa los valores enviados por el usuario o cliente a través del método GET de la página Web, cuando el sistema está en modo manual. Estos valores están orientados a la activación de los actuadores a través de la red.

4.3.5. Proceso de visualización

El código en C para controlar la visualización de los datos se encuentra en el archivo **lcd.c**. En este archivo se definen funciones para el manejo de este elemento como: **lcd_init()** para precisar la forma en que se visualizara la información, otra función importante es **lcd_print_p()** que es la función que permite el envío de los caracteres que se imprimirán en el módulo.

4.3.6. Proceso de control

El sistema implementado permite 2 opciones de control de los equipos de ventilación y calefacción. Los modos son: automático y manual. En el circuito existe un LED que indicará el modo en la que estará trabajando el sistema.

El usuario podrá manipular estos modos desde la página web del sistema. Cuando el sistema trabaje en modo manual el cliente puede encender o apagar los equipos según su conveniencia mientras que en el modo automático los artefactos se encenderán según el rango de trabajo establecido en el firmware.

Cuando la temperatura y humedad medidas no se localicen en el rango, se encenderá el dispositivo según sea el caso durante varios segundos. Transcurrido el tiempo mencionado automáticamente se apagará.

CAPITULO V

ANÁLISIS Y RESULTADOS

Finalmente, se tienen que realizar las pruebas respectivas para determinar si el circuito cumple con los objetivos planteados. Antes de la implementación del mismo en hardware se debe comprobar su funcionamiento por software para evitar daños a los dispositivos adquiridos para posteriormente hacer la placa de circuito impreso.

El proyecto utiliza una fuente de alimentación de 110 V para los actuadores y 5 V para los demás elementos (microcontrolador, lcd, etc.). Para conectar el circuito a la red se puede utilizar como medio de transmisión cable UTP cat 5 con un conector RJ45.

En el navegador o Browser digitamos la dirección IP del servidor que se tenga configurado y a continuación podemos realizar el monitoreo de los sensores de humedad y temperatura, así como el estado de los actuadores.

A continuación se explicará todo el procedimiento efectuado para el análisis y verificación correcto del proyecto.

5.1. Medición de Temperatura y humedad

Para comprobar la veracidad de los valores medidos por el dispositivo implementado se puede recurrir a comparar los resultados con un termómetro (para medir la temperatura) y un Higrómetro (para medir la humedad), sin embargo no es habitual encontrar equipos que midan humedad.

El equipo comercial con el que se realizó las comparaciones fue el DAP-II (Data Alarm Processor II) Serie 2005-3471-E del fabricante DATA AIRE INC (Figura V.41).

Las características de este equipo son:

- Compresores Scroll para mayor confiabilidad y operación silenciosa.
- Humidificador generador de vapor para fácil servicio y costo efectivo.
- Microprocesador DAP-II asegura eficiente operación y comunicaciones.
- Motor con bandas de velocidad variable.
- Calentador eléctrico.
- Filtros de aire con eficiencia del 30%.
- Turbinas centrífugas con doble entrada de aire.
- El sistema de control no solo controla sino monitorea temperatura, humedad, flujo de aire y limpieza.
- Ajuste de valores nominales, configuración y valores límites.
- Las alarmas se visualizan y se mantienen activas.
- Historial de alarmas a prueba de borrado eléctrico.

La DAP-II esta diseñada para necesidades de enfriamiento en cuartos de cómputo en áreas pequeñas, aplicaciones de cuartos de telefonía, Web hosting, áreas de servidores y routers así como cuartos de baterías. Acceso sólo frontal para servicio.

Servicio de mantenimiento sencillo y refacciones de reemplazo fáciles de conseguir. Con el que se ejecutaron 5 mediciones en diferentes tiempos.



Figura V.41. DAP-II (DATA AIRE INC)

Los resultados obtenidos cada 6 segundos se resumen en la Tabla V. 22

Tabla V.22. Mediciones DAP-II vs Sistema Web Embebido

Medición	Sistema Web Embebido		DAP II	
	temperatura	Humedad	temperatura	humedad
t1	21	38	21	37
t2	20	37	20	37
t3	21	38	21	37
t5	22	37	22	37
t4	21	38	22	38
Promedio (\bar{x})	21	37,6	21,2	37,2

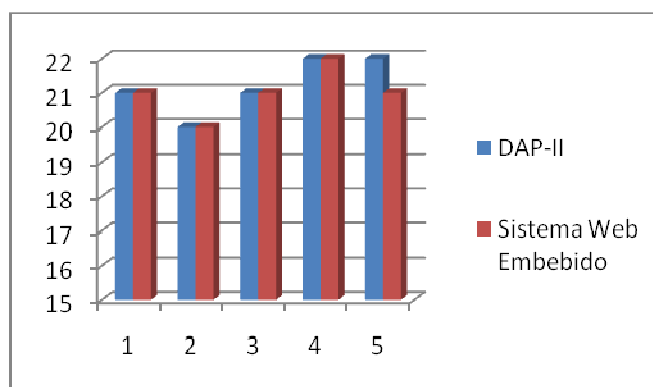


Figura V.42. Mediciones de temperatura DAP II vs Sistema Web Embebido

De la Tabla V.20 y la Figura V.42 se puede observar que no existe mucha diferencia entre los valores medidos de temperatura de los dos equipos, sin embargo es necesario obtener el porcentaje de error $\varepsilon(\%)$ con el siguiente procedimiento:

$$\varepsilon_1(\%) = \frac{|V_m - V_r|}{V_r} * 100 \quad \text{y} \quad \varepsilon(\%) = \frac{\sum \varepsilon_1(\%)}{n}$$

Reemplazando las fórmulas con los valores, los resultados son:

$$\sum \varepsilon_1(\%) = 4,55$$

$$\varepsilon(\%) = 0,91\%$$

De la misma manera se procedió a calcular el error con las mediciones de humedad:

$$\varepsilon_1(\%) = \frac{|38 - 37|}{37} * 100$$

$$\varepsilon_1(\%) = 2,68$$

$$\varepsilon(\%) = \frac{5,26}{5}$$

$$\varepsilon(\%) = 1,052\%$$

Analizando los errores calculados se puede determinar que los porcentajes de error son representativamente bajos, por lo tanto se concluyó que los sensores utilizados para las mediciones son confiables.

5.2. Funcionamiento de los protocolos implementados

Para probar la conectividad se configuró el dispositivo Web embebido con la Dirección IP 10.3.1.1 y la Dirección del cliente 10.3.1.3. Para monitorear y analizar

el tráfico de la red existen varios programas como MRTG, PRTG, Ethereal, Wireshark.

Se utilizó el analizador de tráfico WireShark que se puede acceder gratuitamente en Internet.

5.2.1. Protocolo ARP

De la figura V.43. se observa que el sistema Web responde a una petición ARP (Trama No.1), en este caso la dirección MAC o física del servidor corresponde a 53:45:52:56:45:52 en sistema hexadecimal.

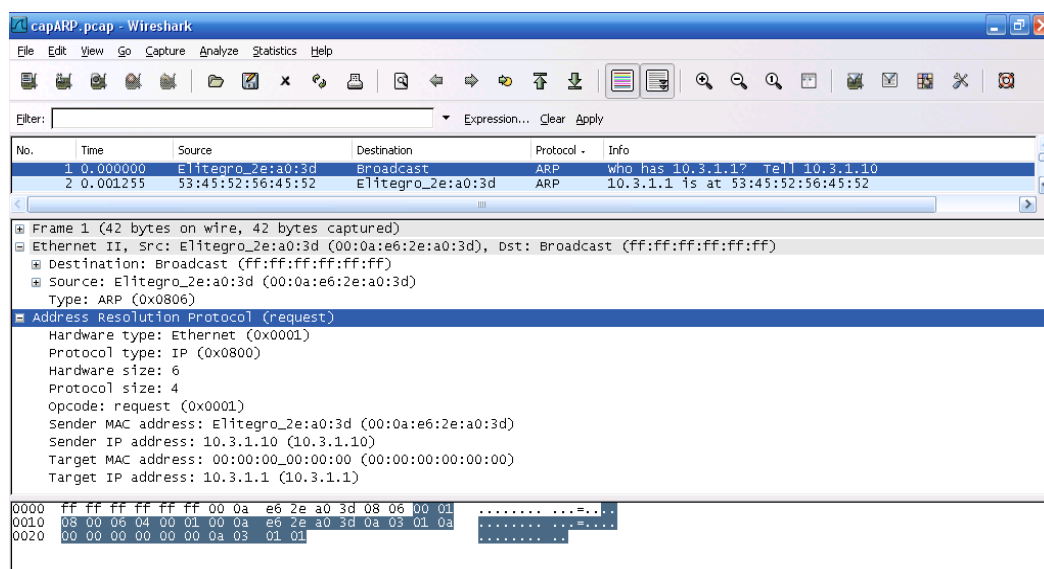


Figura V.43 captura del trafico ARP (Reply)

5.2.2. Protocolo ICMP

Como se observa en la Figura V. 44. se puede concluir que el ICMP con su utilidad Ping funcionan correctamente. El cliente genera 4 peticiones ARP al cual el servidor envía 4 respuestas. Este comando además muestra estadísticas y tiempos aproximados en la respuesta de los paquetes.

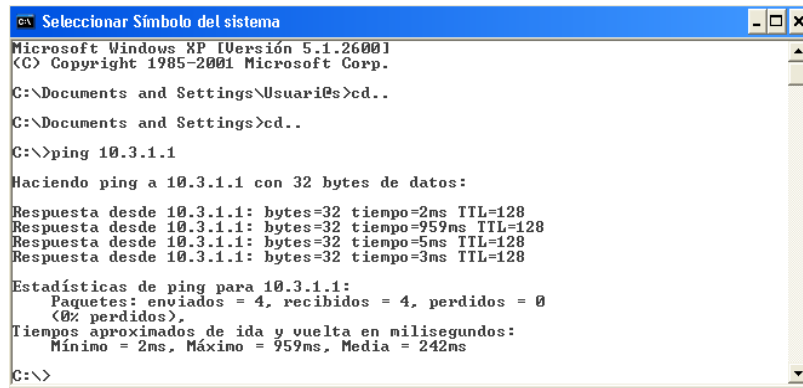


Figura V.44. Conexión verificada mediante Ping

5.2.3. Protocolo TCP

Se decidió implementar el protocolo TCP en lugar del UDP debido a la confiabilidad de este protocolo, pues la confiabilidad era un parámetro fundamental que se debía tomar en cuenta para realizar este proyecto. En el Anexo 6 se puede verificar el tráfico generado entre el cliente y el servidor y en Anexo 7 me muestra un resumen de la captura realizada por el Wireshark durante 25 s de la comunicación entre el sistema Web embebido y un cliente. En la Figura V.45, se puede comprobar mediante las capturas realizadas que se efectúa la comunicación en tres vías.

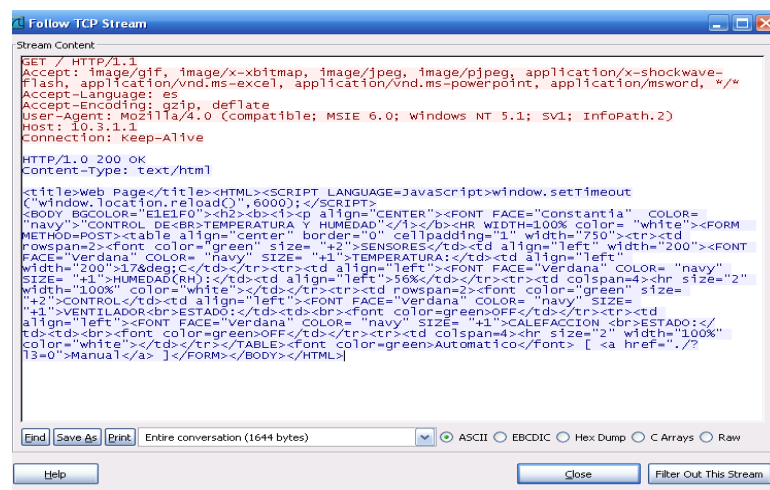


Figura V.45. Flujo TCP

5.2.4. Protocolo HTTP

EL Cliente Web envía la petición hacia el servidor con el método GET (Trama No.6). En el Anexo 4, se observa en la trama No. 7 la respuesta del servidor con el envío de la página Web (Figura V.46), el puerto de origen es el 80 correspondiente al servicio web, y el de destino es un puerto aleatorio que en este caso su valor es 1044. El modo por defecto del sistema es el automático. El usuario podrá manipular los dispositivos de Control cambiando el modo a manual (Ver anexo 5). Para conseguir este cambio se envía la variable I3 con un valor de 0 o 1 para cambiar a automático o manual respectivamente. En el circuito, se podrá comprobar este modo al encenderse el LED3.

De la misma forma para encender o apagar los dispositivos de control las variables a enviarse son L1 y L2 con un valor de 0 o 1 si se requiere cambiar su estado entre OFF y ON. El envío de estos parámetros con sus valores se puede comprobar en el Anexo 6.



Figura V.46. Pagina Web visualizada en el navegador

CONCLUSIONES

- Mediante el estudio de los Sistemas Embebidos se logro la selección adecuada del sistema para el diseño e implementación del prototipo. Para efectuar el procesamiento se optó por el microcontrolador ATMEGA32 y el ENC28J60 para dar al sistema conectividad con la red. Dando como resultado un sistema eficiente y de bajo costo.
- Se implementó el firmware del sistema con funciones y procedimientos enfocados a utilizar el menor espacio de memoria.
- Para darle una parte de confiabilidad al sistema se utilizó el protocolo TCP, ya que implementa control de flujo y solicita la retransmisión de paquetes si ha detectado errores en la información.
- Mediante la comparación de las mediciones de temperatura y humedad entre el sistema con un equipo comercial se determinó que los valores medidos son totalmente confiables.
- El sistema provee el control automático de los actuadores para mantener la temperatura y humedad en un rango de 15-20°C para la temperatura y 40-60% de humedad relativa. Sin embargo, se puede optar por el control manual para encender o apagar los equipos según los requerimientos del usuario.

RECOMENDACIONES

- Si se requiere implementar Bases de datos embebidas por ejemplo para registrar la temperatura y humedad a lo largo del día se recomienda agregar un dispositivo de memoria al sistema, debido a que se utilizó aproximadamente los 30Kbytes de memoria de los 32Kbytes disponibles del microcontrolador.
- El sensor de temperatura no debe ser colocado a una gran distancia, pues la salida es de tipo analógico y la señal podría atenuarse. Por lo tanto se obtendría resultados erróneos en las mediciones.
- Para que dispositivo construido pueda ser utilizado en más aplicaciones se puede programar una rutina para que el usuario pueda ingresar los rangos de temperatura y humedad, sin embargo también se deberá considerar el rango de medición de los sensores.
- Debido a que el sistema no es seguro, se puede implementar algoritmos de encriptación para evitar que usuarios no autorizados accedan a la información mediante herramientas como los analizadores de protocolos.

RESUMEN

En el presente proyecto se realizó un estudio de los Sistemas Web Embebidos con el objetivo de diseñar e implementar un prototipo que permita controlar la temperatura y humedad desde cualquier parte de una red LAN.

Para obtener las mediciones de temperatura se manejó el termistor como sensor y para la humedad se utilizó el sensor HS1101. El estudio permitió la selección del sistema más adecuado para el proyecto. Como dispositivo de procesamiento se optó por el microcontrolador ATMEGA32 de ATMEL por ser más rápido, tener capacidad de memoria y bajo costo. Para lograr la comunicación entre el sistema con la red LAN se agregó el módulo ENC28j60. En el microcontrolador se programó el firmware del sistema en lenguaje C en el software AVR Studio 4. El código incluye la implementación de parte del stack TCP/IP. Se eligió utilizar TCP en lugar del UDP como protocolo de transporte porque TCP es confiable. La página Web diseñada se envía con las mediciones de temperatura, humedad y el estado de los dispositivos de control.

Para comprobar la veracidad de los valores medidos se comparó el sistema con un modelo comercial. El error calculado fue de 0.91% para la medición de la temperatura y 1.05% para la medición de la humedad. Se utilizó un analizador de protocolos, Wireshark, para monitorear el tráfico de la red. El número promedio de paquetes generados por la comunicación entre el sistema y el cliente es 1,162 packets/sec. De esta forma se comprueba la eficiencia del sistema Web Embebido implementado.

SUMMARY

This project conducted the study Embed WEB systems for designing and implementing a prototype that allows the temperature and humidity control from anywhere on a LAN.

For the temperature measurements was used as a thermostat sensor and humidity sensor using a HS1101. The study allowed the selection of the most appropriate system for the project. As device processing was chosen by the micro controller ATMEL ATMEGA 32 to be faster, have more memory capacity and low cost. To achieve communication between the systems with the LAN module is added ENC2J60. The microcontroller is programmed in the system firmware in C language in the AVR Studio 4 software.

The code includes the implementation of part stack TCP/IP. It chose to use TCP instead of UDP as a transport protocol because TCP is reliable. The website is designed to send the measurements of temperature, humidity and status control devices.

The website is designed to send the measurements of temperature, humidity and status control devices. To verify the measured values are compared with the business model. The calculated error was 0.91% for the measurement of 1.05% for temperature and humidity measurement.

We used a protocol analyzer, Wireshark to monitor network traffic. The average number of packets generated by the communication between the system and the customer is 1,162 packets / sec. found in this way the system efficiency WEB Embed implemented.

GLOSARIO

AVR: Familia de microprocesadores RISC de la empresa ATMEL CORP. La arquitectura de los AVR fue concebida por dos estudiantes en el Norwegian Institute of Technology, y posteriormente refinada y desarrollada en Atmel Norway, la empresa subsidiaria de ATMEL, fundada por los dos arquitectos del chip.

Buffer: Memoria intermedia que se utiliza como memoria de datos temporal durante una sesión de trabajo

Ethernet: especificación de red de área local (LAN) desarrollada en 1976 por Xerox, en cooperación con DEC e Intel, originalmente para conectar los miniordenadores del Palo Alto Research Center (EEUU). Se trata de una red muy difundida, de la cual se derivó la norma (o estándar) IEEE 802.3 para redes de conexión.

ENC28J60: Circuito Integrado perteneciente al fabricante MICROCHIP TECHNOLOGY, que sirve como medio de comunicación entre un microprocesador y una red Ethernet.

Firmware: Es el software que está dentro del hardware. Se refiere a los programas grabados en memorias no volátiles cuya función es asegurar su correcto funcionamiento.

RFCs: Contienen las especificaciones formales de los protocolos de comunicación de datos y los recursos que describen el uso de los protocolos.

Sistema embebido: Son sistemas orientados a cumplir con una o varias tareas específicas. Dando paso a la utilización óptima de recursos.

Termistor: El termistor compuesto de una mezcla sintetizada de óxidos metálicos es una resistencia termoestable que puede ser utilizada tanto en circuitos de corriente continua como en alterna.

Anexos

ANEXO 1

TERMISTOR NTC

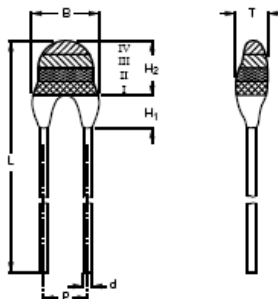
Los termistores también se pueden encontrar en el mercado con la denominación **NTC** (Negative Temperature Coefficient) habiendo casos especiales de coeficiente positivo cuando su resistencia aumenta con la temperatura y este tipo de termistor se le denomina **PTC** (Positive Temperature Coefficient).

TERMISTOR NTC DE 10 K

Los termistores NTC poseen elevadas resistencias a baja temperatura, pero sus resistencias disminuyen exponencialmente a medida que crece la temperatura.

PARAMETER	VALUE
Resistance value at 25 °C	3.3 Ω to 470 kΩ
Tolerance on R ₂₅ -value	±2%; ±3%; ±5%; ±10%
Tolerance on B _{25/85} -value	±0.5% to ±3%
Maximum dissipation	500 mW
Dissipation factor δ (for information only)	7 mW/K 8.5 mW/K (for 640..338 to 689)
Response time	1.2 s
Thermal time constant τ (for information only)	15 s

DIMENSIONES EN mm



2322 640 6.338 to 6.474.

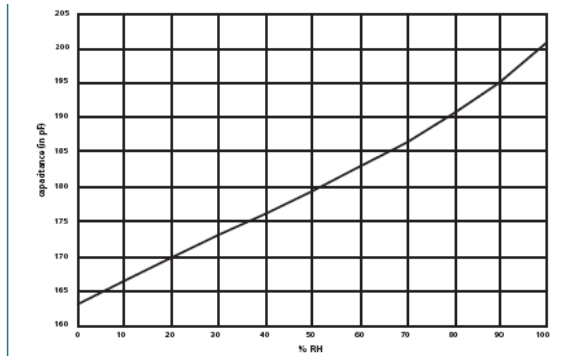
CODE NUMBER 2322 640	B _{max}	d	H ₁		H ₂ max	L	P	T _{max}
			MIN.	MAX.				
6.338 to 6.221	5.0	0.6 ±0.06	1.0	4.0	6.0	24 ±1.5	2.54	4.0
6.331 to 6.474	3.3 ±0.5	0.6 ±0.06	-	2.0 ±1.0	6.0	24 ±1.5	2.54	3.0

ANEXO 2

HS1101

Este es un sensor capacitivo, que varía su constante dieléctrica en función de la Humedad Relativa.

CURVA RESPUESTA TÍPICA DE HS1101 RESPECTO A LA HUMEDAD



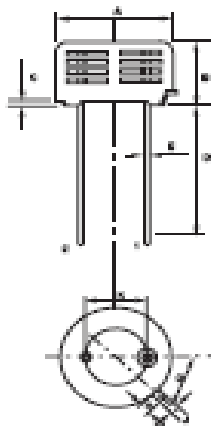
La frecuencia de la medida:
10 KHz
Ta = 25°C

Polinomio de Respuesta:

$$C(pf) = C@55\% * (1.2510^{-7} RH^3 - 1.3610^{-5} RH^2 + 2.1910^{-3} RH + 9.010^{-1})$$

Características

- Intercambiabilidad total, en condiciones normales no requiere calibración
- Alta fiabilidad y largo tiempo de estabilidad
- Estructura de polímero sólido patentada
- Apropiado para circuitos lineales o de impulsos
- Tiempo de respuesta rápido
- Para conseguir una buena respuesta durante todas la medidas, conectar siempre el encapsulado (Pin 2) a la tierra del circuito.

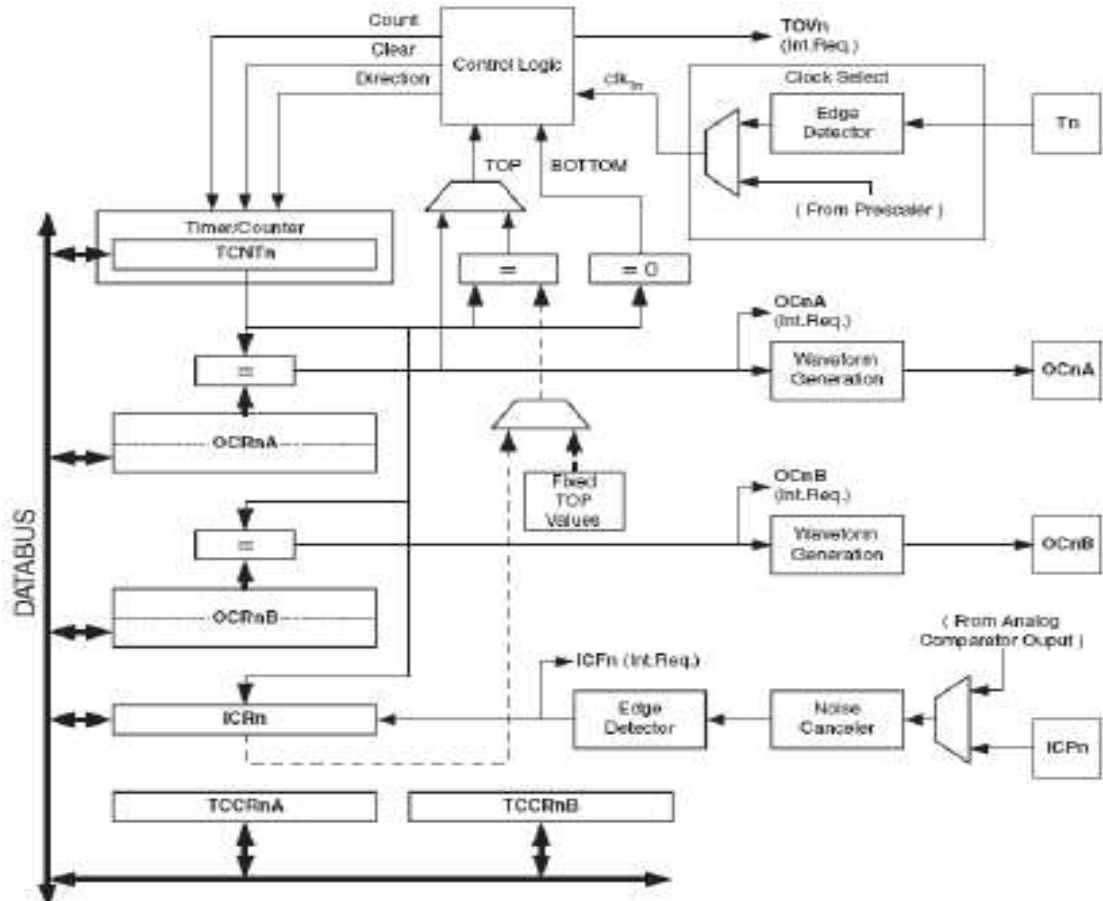


Dim	Min (mm)	Max (mm)
A	9.70	10.20
B	5.70	6.20
C	0.40	0.60
D	12.00	14.00
E	0.40	0.50
G	45° BCS	
H	0.70	1.10
J	0.70	0.90
K	4.83	5.33

Dimensiones del sensor HS1101

ANEXO 3

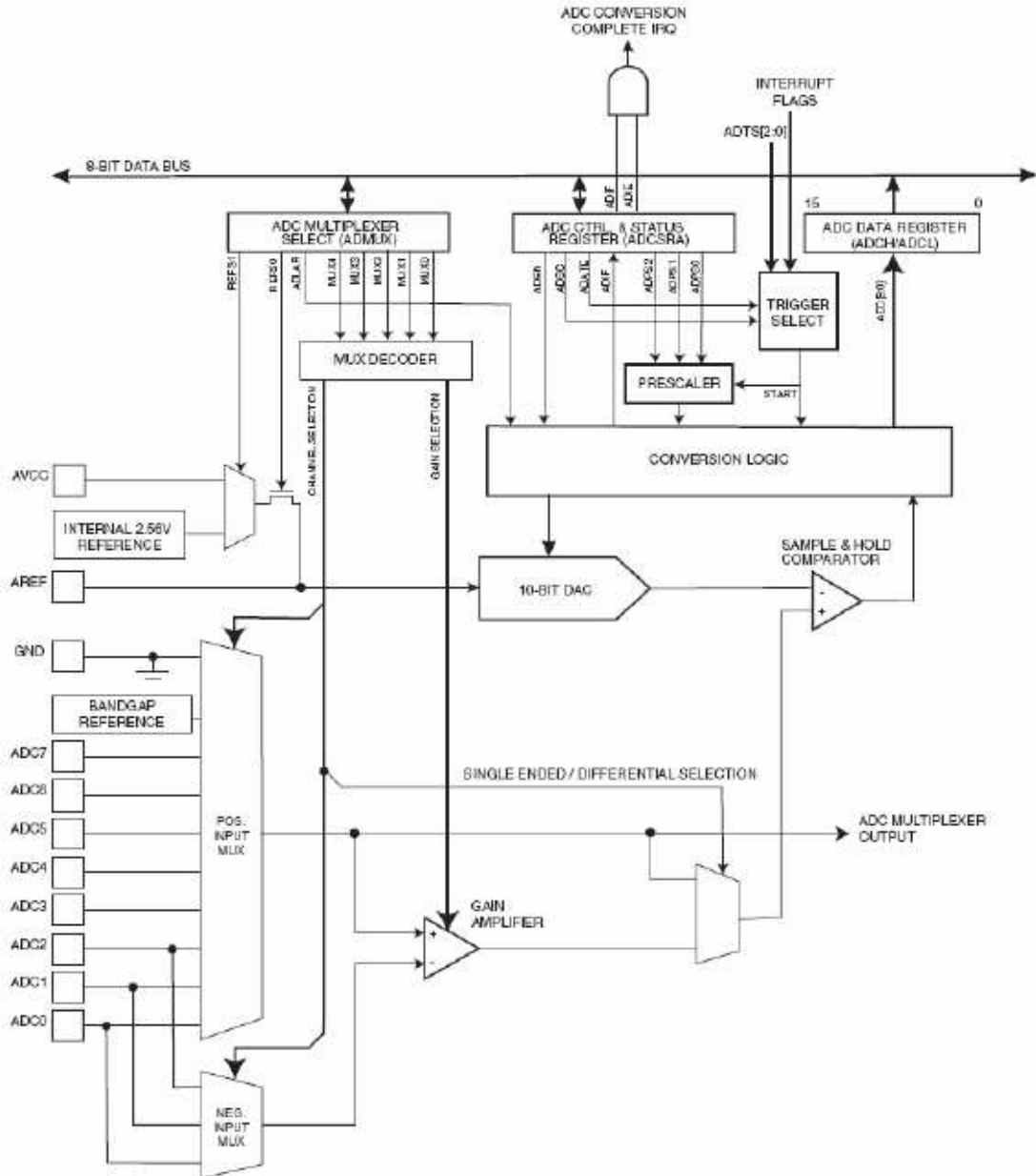
DIAGRAMA DE BLOQUES DEL TEMPORIZADOR / CONTADOR 1



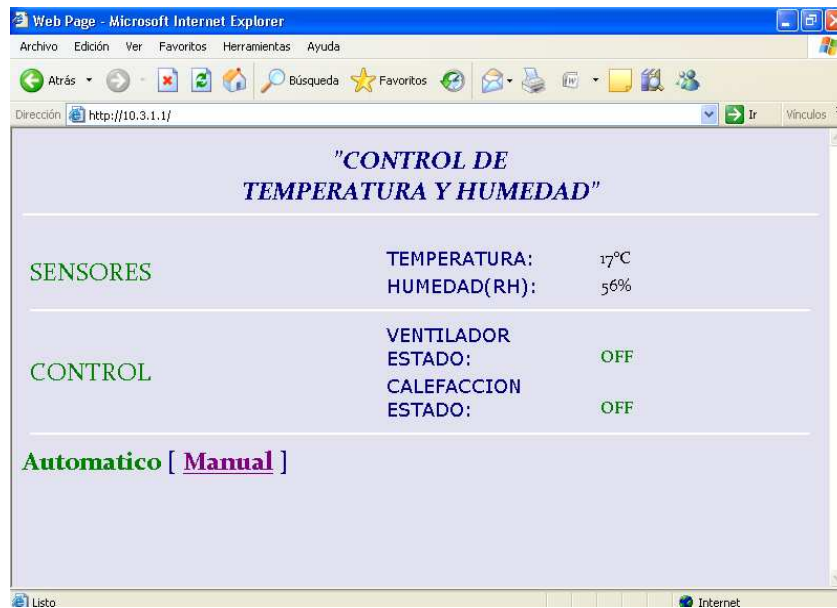
ANEXO 4

DIAGRAMA DE BLOQUES DE CONVETIDOR ANALOGICO DIGITAL

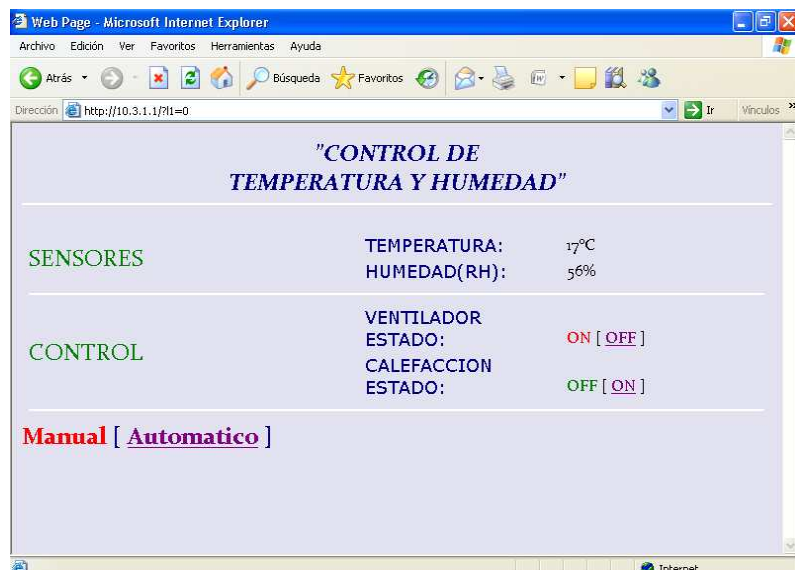
DEL ATMEGA32



ANEXO 5

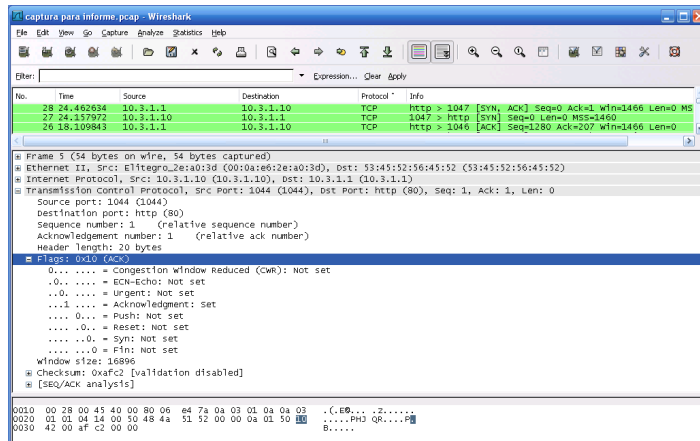


Página Web con el Sistema en modo Automático

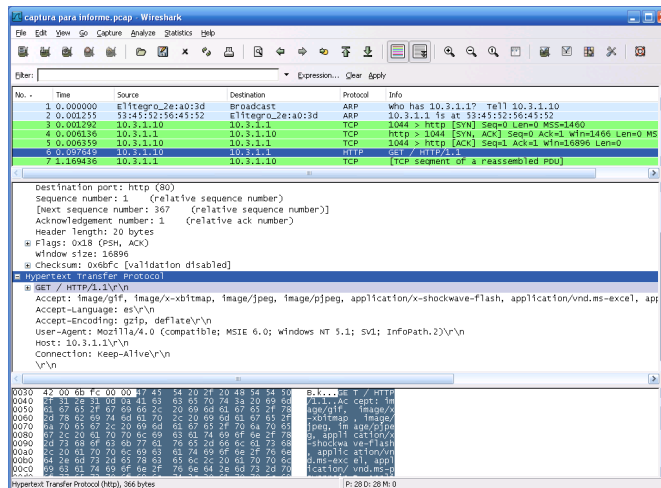


Página Web del sistema en modo manual controlando los estados de los equipos por software

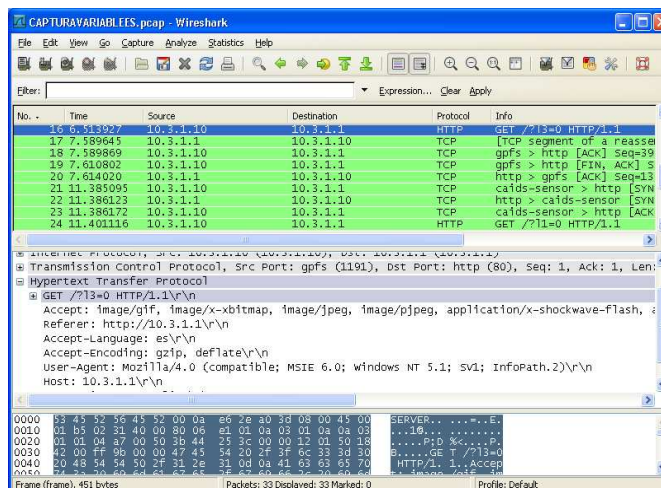
ANEXO 6



Captura del tráfico TCP: Comunicación en tres vías



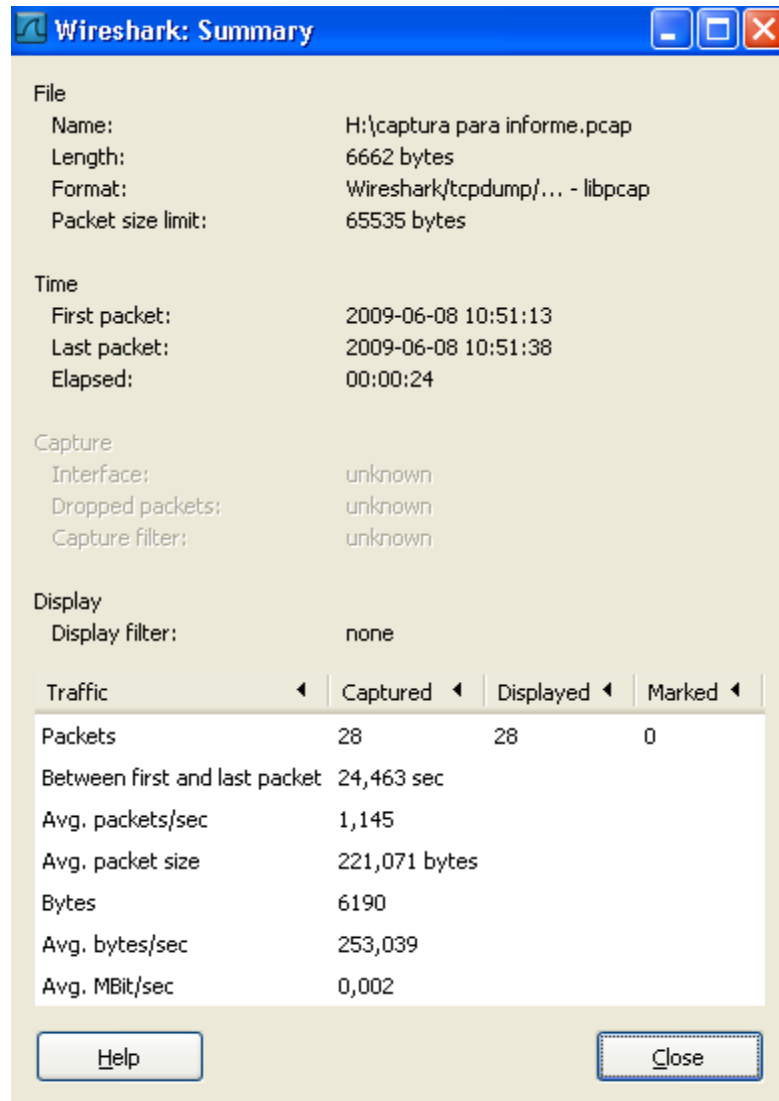
Captura del tráfico HTTP y TCP



Envío de variables mediante el método GET

ANEXO 7

Resumen Estadístico de la comunicación entre el Sistema Web embebido y un cliente



The image shows a screenshot of the 'Wireshark: Summary' window. The window title is 'Wireshark: Summary'. It displays various statistics for a capture file named 'H:\captura para informe.pcap'.

File

- Name: H:\captura para informe.pcap
- Length: 6662 bytes
- Format: Wireshark/tcpdump/... - libpcap
- Packet size limit: 65535 bytes

Time

- First packet: 2009-06-08 10:51:13
- Last packet: 2009-06-08 10:51:38
- Elapsed: 00:00:24

Capture

- Interface: unknown
- Dropped packets: unknown
- Capture filter: unknown

Display

- Display filter: none

Traffic

	Captured	Displayed	Marked
Packets	28	28	0
Between first and last packet	24,463 sec		
Avg. packets/sec	1,145		
Avg. packet size	221,071 bytes		
Bytes	6190		
Avg. bytes/sec	253,039		
Avg. MBit/sec	0,002		

Buttons: Help, Close

BIBLIOGRAFÍA

LIBROS

- **CARRANZA LUJAN, J.** Implementación y Configuración de las Redes. México: Megabyte, 2006. pp. 1-551.
- **LABROSSE, J.** Embedded Software. Estados Unidos: Newnes, 2008. pp. 169-331. (Doc. PDF).
- **STALLING, W.** Comunicaciones y Redes de Computadores. 7ma ed. Madrid: Pearson, 2004. pp. 587-805.
- **TANENBAUM, A.S.** Redes de Computadoras. 4ta ed. Madrid: Prentice Hall, Hispanoamericana, 2003. pp. 1-891.
- **WILMSHURST, T.** Systems with PIC Microcontrollers. Estados Unidos: Newnes, 2007. pp. 1-22 (Doc. PDF).

BIBLIOGRAFÍA DE INTERNET

- **DATASHEET ATMEGA32**

<http://www.avrfreaks.net/index.php?module=Freaks%20Files&func=viewFile&id=1238&showinfo=1>

2008-10-08

- **DATASHEET ENC28J60**

<http://ww1.microchip.com/downloads/en/DeviceDoc/39662a.pdf>

2008-10-20

- **DATASHEET HS1101**

<http://www.humirel.com/product/fichier/HS1101-HS1100.pdf>

2008-09-05

- **HUMEDAD RELATIVA**

<http://www.fao.org/wairdocs/x5403s/x5403s09.htm>

2008-09-01

- **INTERFAZ SPI**

<http://es.emcelettronica.com/05-la-interface-spi.html>

2008-11-22

- **PROGRAMACIÓN DE BITS FUSES**

<http://www.scienceprog.com/programming-avr-fuse-bits-oscillator-settings.html/>

2008-11-28

- **SENSOR DE TEMPERATURA**

http://www.sapiensman.com/medicion_de_temperatura/sensores_de_temperatura.htm

2008-08-10

- **TIPOS DE ACTUADORES**

http://cfievadolid2.net/tecno/cyr_01/robotica/sistema/actuadores.htm

2009-04-03

A DIOS por brindarme salud para poder llevar a cabo este proyecto de mi vida; de manera especial dedico la presente investigación a MIS PADRES Aida y Gonzalo por guiarme incondicionalmente en todas mis etapas de desarrollo humano y profesional; a MIS HERMANOS Bertha y Alex por apoyarme con su confianza día a día y por el tiempo que han sacrificado, por la comprensión y cariño brindados, porque de una u otra manera la familia es quien siempre nos acompaña en los buenos o malos momentos.

Franklin Bravo

A la Escuela Superior Politécnica de Chimborazo ESPOCH, la cual me abrió sus puertas brindándome la oportunidad de obtener los conocimientos necesarios para llegar a obtener mi título.

A los Señores Profesores de la Facultad que en su momento supieron orientar mi formación académica y de manera muy especial al ING.

Hugo Moreno por su valioso tiempo y su acertada dirección en el desarrollo de esta investigación.

Finalmente gracias a DIOS, por permitirme tener salud y así lograr culminar con este objetivo profesional.

Franklin Bravo

