

Diseño y construcción de un dispositivo de desarrollo de microcontroladores para el Laboratorio de Sistemas Microprocesados, utilizando la tecnología Bluetooth para la descarga de los programas.

José Javier De la Torre Guzmán, Ing. Jaime E. Velarde.

Ingeniería Electrónica y Telecomunicaciones, Escuela Politécnica Nacional

Quito, Ecuador

Abstract- El aprendizaje de los sistemas microprocesados se empieza conociendo la estructura interna del microcontrolador, para luego aprender los mnemónicos que nos permita realizar las aplicaciones que puedan resolver los problemas, pero se debe tener en cuenta que los dispositivos que se conecta a un procesador han evolucionado y el estudiante debe estar capacitado en su utilización, por lo cual este proyecto da una alternativa de manejo de otras tecnologías, como también facilitar la descarga del código fuente de la práctica mediante una comunicación inalámbrica, con esto se pretende un mejor aprendizaje de los microcontroladores facilitando al estudiante el desarrollo de las prácticas.

SITUACIÓN ACTUAL EN LA IMPLEMENTACIÓN DE LAS PRÁCTICAS EN EL LABORATORIO DE SISTEMAS MICROPROCESADOS

Es importante indicar la actual implementación de las prácticas de laboratorio, para lo cual se realiza una encuesta para conocer la opinión del estudiante sobre la implementación en el laboratorio en base a sus fortalezas como debilidades que luego se toman en cuenta para el diseño del dispositivo

A. Situación a nivel de Hardware

Para la realización de las prácticas se necesitan dentro del hardware varios dispositivos que se detallan a continuación:

- Microcontrolador
- Programador
- Protoboard
- Dip switch
- Display de 7 segmentos
- Teclado
- Otros

Todos estos elementos son indispensables para la realización de las prácticas de laboratorio, cabe recalcar que la interconexión de los mismos depende del estudiante, lo que quiere decir que el correcto funcionamiento depende si estos fueron conectados correctamente.

B. Situación a nivel de Software

La parte de software o programación se define como la creación de una o varias rutinas dentro de un programa el cual se va ejecutar por medio del microcontrolador.

La programación esté enfocada a aprender los medios para la utilización de las diferentes herramientas que da el microcontrolador como son:

- Pórticos de E/S
- Procesador aritmético
- Manejo de memorias
- Conversor análogo-digital
- Timers
- Interrupciones
- Comunicación serial

El lenguaje de programación utilizado en el laboratorio es Ensamblador, el cual se lo implementa para traducir las instrucciones a lenguaje de máquina

Para la elaboración del programa se sigue los pasos que se detallan:

1) *Definición del problema:* este se desenvuelve en la comprensión que permita plantear la aplicación que se quiere efectuar en el laboratorio.

2) *Elaboración del Algoritmo:* es indispensable que el estudiante realice una o varias rutinas que resuelva el problema de la práctica.

3) *Escritura del programa en mnemónicos:* existe varios lenguajes de programación entre los que están: C, C++, Basic, Ensamblador, para lo cual se usa un cierto

número de mnemónicos para traducir a lenguaje de máquina.

4) *Traducción del programa en lenguaje de máquina*: se utiliza compiladores que cree este archivo, para el caso del laboratorio se utiliza el software AVR Studio 4

C. Encuesta realizada a los estudiantes de Laboratorio de Sistemas Microprocesados

La encuesta a los estudiantes de laboratorio de sistemas microprocesados permite detallar los mecanismos actuales de la implementación de la prácticas en base sus fortalezas como debilidades que se toman en cuenta para el diseño del dispositivo.

A continuación se pasa a mostrar los resultados de la encuesta:

1) ¿Qué grado de aceptación tiene la forma de desarrollar actualmente las prácticas de Laboratorio?

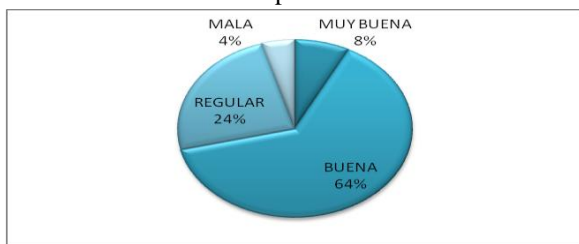


Fig. 1 Gráfica de la tabulación a la pregunta No. 1 de la encuesta

2) ¿Ha tenido usted algún problema con el hardware que implementa en el Laboratorio?

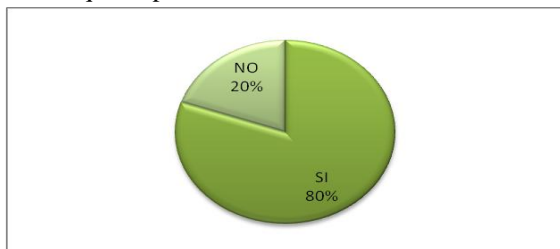


Fig. 2 Gráfica de la tabulación a la pregunta No. 2 de la encuesta

3) ¿El laboratorio cuenta con los suficientes programadores del microprocesador?

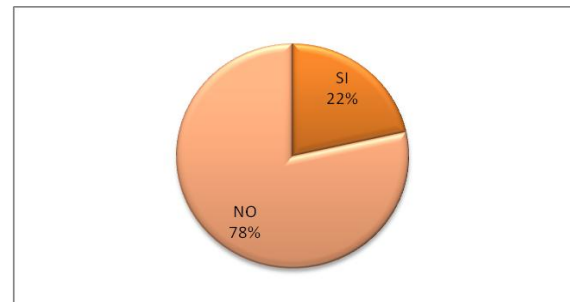


Fig. 3 Gráfica de la tabulación a la pregunta No. 3 de la encuesta

4) ¿Qué tipo de problemas tiene al realizar las prácticas?

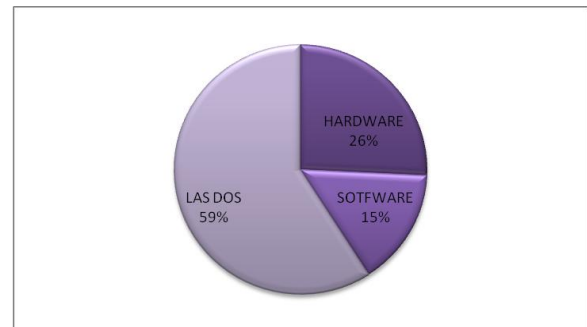


Fig. 4 Gráfica de la tabulación a la pregunta No. 4 de la encuesta

5) ¿Puede realizar modificaciones del Software dentro de las prácticas de manera eficiente?

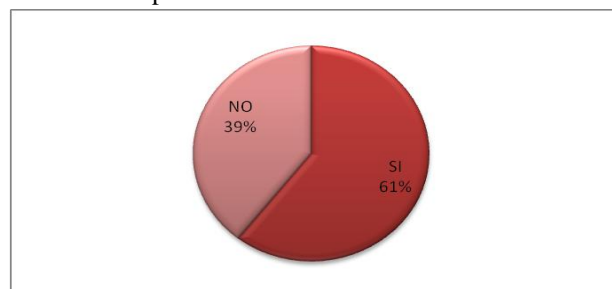


Fig. 5 Gráfica de la tabulación a la pregunta No. 5 de la encuesta

6) ¿Cree que es necesario disponer de algún dispositivo que sirva de programador?

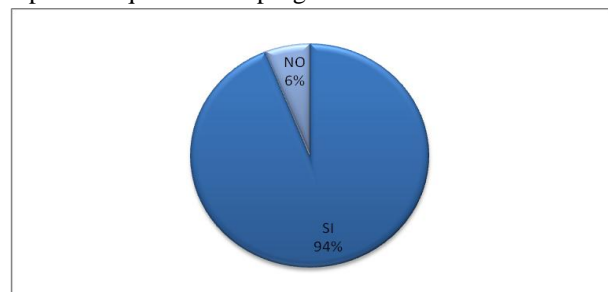


Fig. 6 Gráfica de la tabulación a la pregunta No. 6 de la encuesta

7) ¿Cree que es conveniente desarrollar un dispositivo que facilite la implementación de las prácticas?

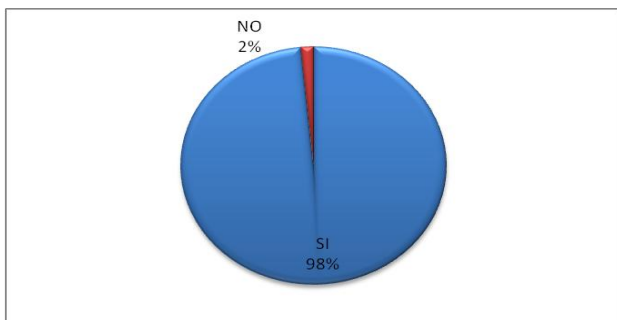


Fig. 7 Gráfica de la tabulación a la pregunta No. 7 de la encuesta

8) ¿Le gustaría un sistema que evite la adquisición del microprocesador y solo sea necesario el programa de la práctica?

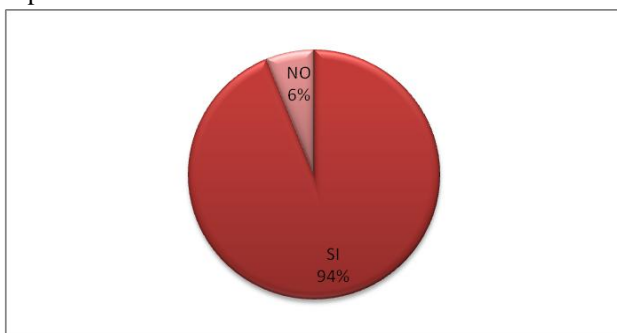


Fig. 8 Gráfica de la tabulación a la pregunta No. 8 de la encuesta

9) ¿Le gustaría que el dispositivo contenga un sistema de salida gráfica?

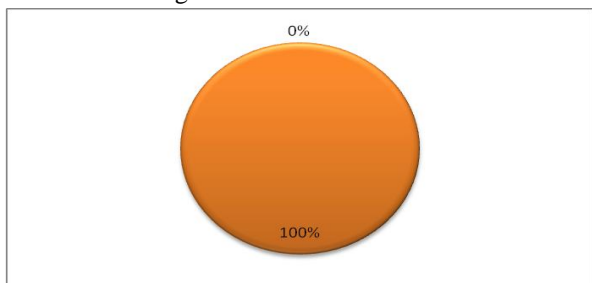


Fig. 9 Gráfica de la tabulación a la pregunta No. 9 de la encuesta

10) ¿Le gustaría que el dispositivo contenga un dispositivo que facilite el ingreso de datos?

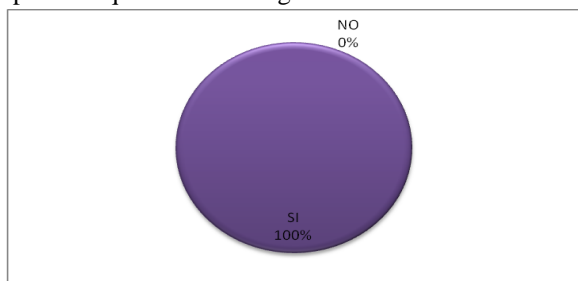


Fig. 10 Gráfica de la tabulación a la pregunta No. 10 de la encuesta

11) Cree que es necesario que el sistema permita cargar el programa de manera inalámbrica en las prácticas de Laboratorio?

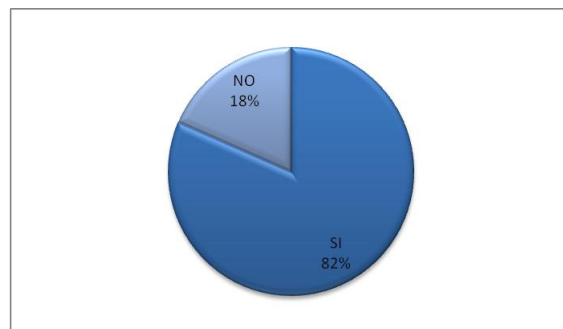


Fig. 11 Gráfica de la tabulación a la pregunta No. 11 de la encuesta

12) ¿le gustaría un software que facilite el envío del programa desde la PC hacia el dispositivo de desarrollo?

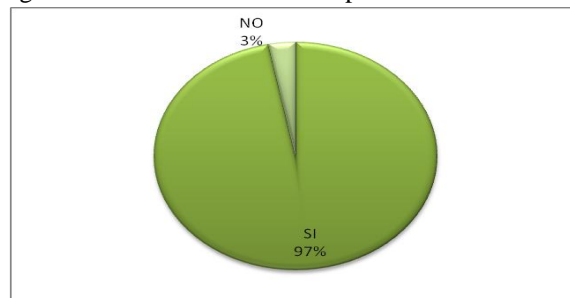


Fig. 12 Gráfica de la tabulación a la pregunta No. 12 de la encuesta

13) ¿Cree que obtendría mayor beneficio en la implementación de las prácticas con este dispositivo?

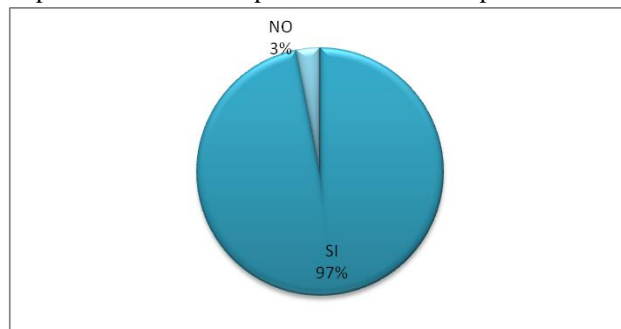


Fig. 13 Gráfica de la tabulación a la pregunta No. 13 de la encuesta

D. Evaluación de la información

Una vez evaluada la información se concluye que la implementación de las prácticas en el laboratorio tiene varias falencias, por lo cual se realiza el diseño y construcción de un dispositivo (PDAT164P) que sirve para que el estudiante en el desarrollo de la práctica resuelva los problemas del hardware como: cortocircuitos de los elementos, la adquisición de elementos como son el microcontrolador, permitiendo que no sea necesario llevar a laboratorio ningún sistema que sirva de programador.

Con la creación de este dispositivo se logra que el estudiante desarrolle nuevas habilidades en la programación sobre los microcontroladores enfocándose a la parte de software de la práctica, sin que deje de lado el

hardware, ya que es indispensable para el desarrollo de sistemas

I. DISEÑO Y CONSTRUCCIÓN DEL DISPOSITIVO

En lo referente al diseño de hardware del PDAT164P se debe indicar que se utiliza los siguientes elementos: microcontrolador Atmega 164P, un teclado 4X4, una LCD gráfica (GLCD), módulo Bluetooth, una fuente de voltaje.

En lo concerniente al software, se establece las características que el microcontrolador en la memoria Flash, principalmente en la parte de Bootloader donde va a residir el programa para auto programación del microcontrolador.

Bootloader es un programa sencillo que proporciona un soporte de auto programación al microcontrolador.

A. Diseño del Hardware

En la figura 14 se puede observar el diagrama de bloques de PDAT164P.

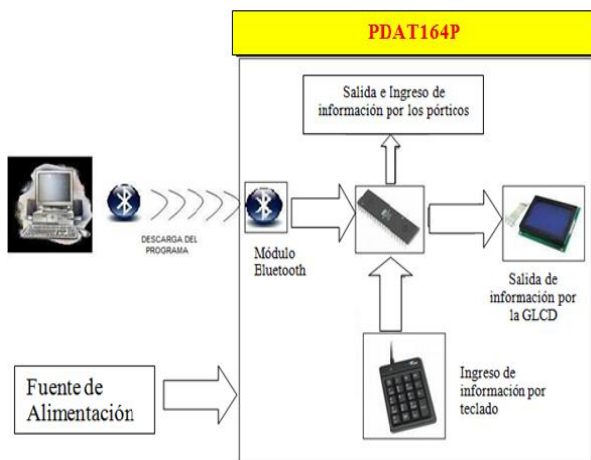


Fig. 14 Diagrama de bloques de PDAT164P

A continuación se procede a describir los elementos que constituye el PDAT164P

1) *Visualización:* Para la visualización de los resultados de las prácticas se utiliza la GLCD JHD1284E de 128X64, entre sus principales características están:

- 128x64 puntos.
- Modo de visualización (Azul/Amarillo).
- 8 líneas de datos paralelos.
- Visualización de la dirección en Tiempos de reloj.

Las líneas de datos de la GLCD se conectan directamente al pñrtico C del microcontrolador y las

líneas de control RS, R/W, E, CS1 y CS2 se conectan a las terminales 5, 6, 7, 4, 3 de pñrtico D respectivamente, la línea RST de la GLCD va al reset del microcontrolador, VEE y V0 va al potenciómetro para el ajuste de contraste, el resto de líneas son para alimentación del GLCD, por lo cual van a VCC y GND. En la figura 15 se pude ver la interconexión de los elementos.

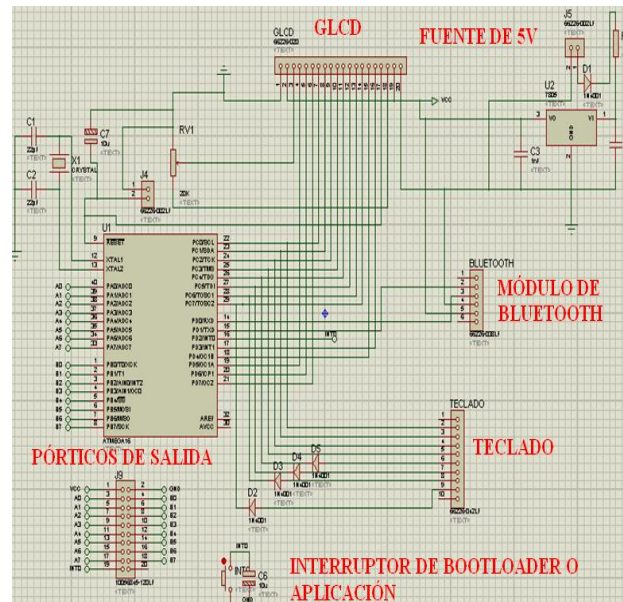


Fig. 15 Diagrama de interconexión de los elementos que conforma el PDAT164P

2) *Teclado 4X4:* en el diseño se maneja el mismo pñrtico que utilizamos para la conexión de las líneas de datos de la GLCD, el pñrtico C; para protección de la GLCD se conecta 4 diodos IN4007 a las líneas de barrido del teclado para evitar cortocircuitos por la pulsación de dos o más teclas a la vez.

3) *Módulo de Bluetooth:* para su aplicación en el presente diseño se utiliza el módulo de comunicación de Bluetooth RN-41, este módulo permite el envío y recepción de datos desde el microcontrolador a la PC. A continuación se describen las características del módulo RN-41:

- Módulo de Bluetooth Clase I
- Bluetooth v2.0 + EDR de apoyo
- Bajo consumo de energía: Promedio de 30 mA
- UART (SPP o HCI) y USB (sólo HCI).
- En SPP tiene velocidades de datos: 240 kbps (esclavo) y 300kbps (maestro)
- Memoria Flash de 8Mb
- Voltaje de funcionamiento: 4.5 – 6 V
- Antena RF integrada en el Chip con amplificador de alta potencia
- Dimensiones 13.4mm x 25.8 mm x2mm
- Puertos UART : TX, RX, RTS, CTS para SPP



Fig. 16 Módulo Bluetooth RN-41

4) *Mini USB-Bluetooth*: Este módulo es de simple operación, ya que descubre y aparece en la pantalla del computador otros dispositivos Bluetooth, en este caso para conectarse con el módulo Bluetooth RN-41.



Fig. 17 Mini USB Bluetooth 2.0

5) *Comunicación entre la PC y el dispositivo de desarrollo*: Para la comunicación entre la PC y el PDAT164P se utiliza un adaptador mini USB-Bluetooth que se conecta a un puerto USB del computador, este adaptador transmitirá vía Bluetooth los datos hacia el módulo Bluetooth RN-41 para que este transforme la información a señales RS-232 para la descarga del programa hacia el microcontrolador ATMEGA 164P, el cual procesa la información. En la figura 18 se muestra el proceso por el que tiene que pasar la señal para llegar al dispositivo y viceversa.

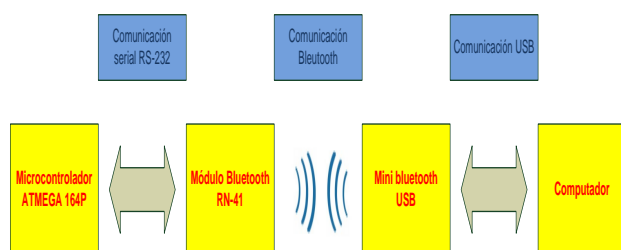


Fig. 18 Diagrama de bloques de la comunicación

Para la comunicación entre dispositivos Bluetooth tenemos varias clasificaciones como son: Clase 1, Clase 2 o Clase 3, en relación a su potencia de transmisión y a su alcance, siendo totalmente compatibles los dispositivos de una clase con los de las otras.

TABLA 1
CARACTERÍSTICAS DE LAS DIFERENTES CLASES
BLUETOOTH

Clase	Potencia máxima permitida (mW)	Potencia máxima permitida (dBm)	Rango (aproximado)
Clase 1	100 mW	20 dBm	~100 metros
Clase 2	2.5 mW	4 dBm	~10 metros
Clase 3	1 mW	0 dBm	~1 metro

El dispositivo de desarrollo trabaja con el módulo RN-41 que es un módulo de clase 1 (V.2.0+EDR), lo que quiere decir que tiene una velocidad máxima de transmisión de 3Mbits y al ser clase 1 tiene un alcance aproximado de 100 m; que es suficiente para el diseño del dispositivo debido a que la comunicación es punto a punto entre el prototipo y la computadora, y no es necesario grandes distancias para la comunicación para la descarga del programa.

6) *Fuente de alimentación*: El PDAT164P va a ser alimentado con 5V. Por tal razón se va a utilizar una batería que entrega 12V, estos deben ser regulados a 5V, para lo cual utiliza un regulador de voltaje fijo LM7805 con un capacitor de entrada y dos de salida para el rizado.

7) *Construcción de la placa del PDAT164P*: Para la construcción del PDAT164P se realiza la placa del dispositivo como de la fuente de alimentación.

Una vez realizado el diagrama circuital del PDAT164P, se transfiere a una placa de fibra de vidrio a doble lado, las cuales deben coincidir para que no haya errores para realizar los huecos, y finalmente soldar los elementos que compone el dispositivo. En la Figura 21 se puede observar la placa del PDAT164P con los elementos que conforma ya montados en la misma.

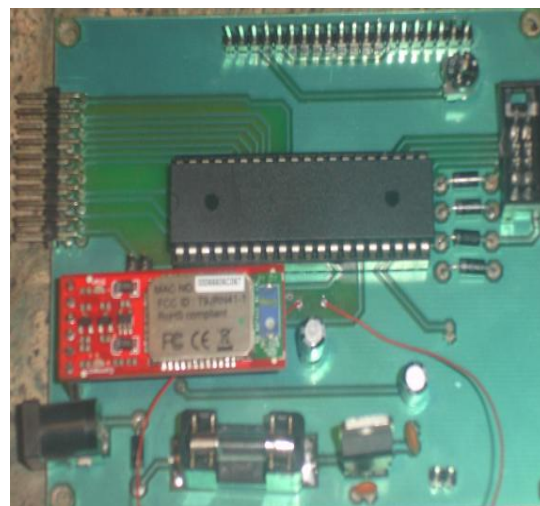


Fig. 19 Diagrama del circuito de alimentación

Se necesita de una caja contenedora para el PDAT164P. Se Utiliza una caja de acrílico de medidas 19x10x6 cm ya de ahí se elabora el PBC (Printed circuit board) o placa de circuito impreso con todos sus componentes el mismo que va a ir sujetado en el interior de la caja. En la parte delantera de la caja, tomando en cuenta las medidas, se monta la GLCD y el teclado 4x4 y un lado lateral las salidas de pórtilo A y B de microcontrolador y el conector de la fuente de 5V. En la figura 20 se muestra el PDAT164P ya construido y montado en la caja acrílica con todos los elementos que constituye el dispositivo.



Fig. 20 PDAT164P

B. Diseño del Software

Para el diseño del software se describe las características de la memoria Flash del microcontrolador ATMEGA164P enfocándose en la parte del Bootloader.

Bootloader es un programa sencillo que proporciona un soporte de auto programación. Esto permite realizar aplicaciones flexibles, con actualización del microcontrolador usando el código que reside en la parte de la memoria Flash llamada sección Bootloader.

El programa Bootloader puede ser usado con cualquier interfaz y tecnología disponible, leyendo el código y actualizando la parte de la memoria llamada sección Aplicación. A continuación se describe las dos secciones de la memoria Flash.

1) *Sección Aplicación*: es la parte de la memoria Flash que se usa para guardar el código del programa.

2) *Sección Bootloader*: contiene el software que permite guardar y actualizar el programa de la aplicación.

El microcontrolador soporta dos maneras de leer el código del programa en la memoria, lee mientras se realiza la escritura, y la otra, el microcontrolador se detiene durante la actualización, en cualquiera de las opciones la actualización depende del número de la página de la memoria donde se quiere colocar el código.

El límite entre RWW y NRWW se puede observar en la figura 21.

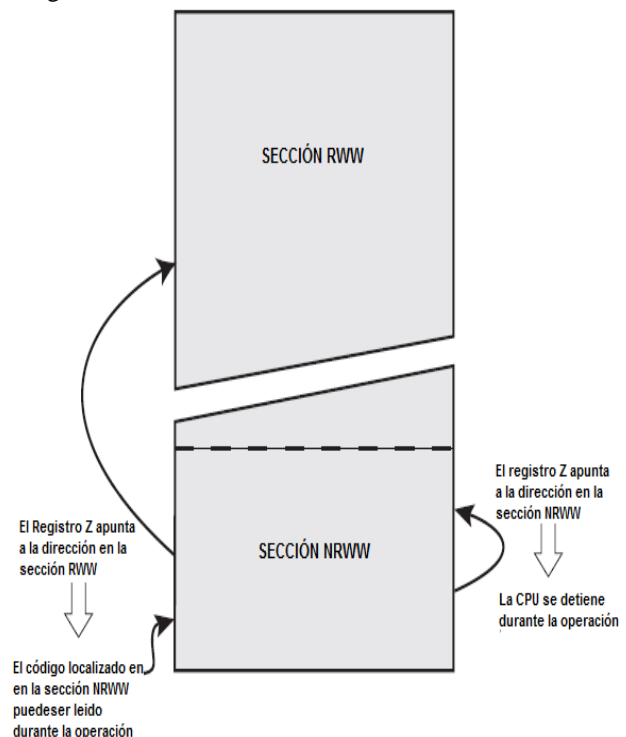


Fig. 21 Secciones RWW vs. NRWW

A continuación se describe las maneras para cargar el código del programa, lectura mientras se escribe (RWW) y no lectura mientras se escribe (NRWW).

1) *Lectura Mientras se Escribe RWW*: Cuando se utiliza la sección RWW para borrar o escribir una página localizada en la misma sección, la sección NRWW puede ser leída.

2) *Sección de No Lectura Mientras se Realiza la Escritura NRWW*: se lee cuando el software Bootloader actualiza la página en la sección RWW. Cuando el código del Bootloader actualiza la sección NRWW, el microcontrolador se detiene durante la operación de borrado de página o escritura de la misma.

La memoria FLASH está organizada por páginas; para el ATmega164P son:

- Tamaño de la FLASH: 16 Kilo bytes = 8 Kilo palabras
- Tamaño de la página: 64 palabras = 128 bytes
- Palabra dentro de la página: PC[5:0] = Z6:Z1
- Número de páginas: 128
- Página: PC[12:6] = Z13:Z7
- Bit más significativo del PC: 12

En el registro Z los bits menos significantes, están direccionando las palabras dentro de la página, mientras los bits más significativos están direccionando las páginas. Esto se muestra en la figura 22.

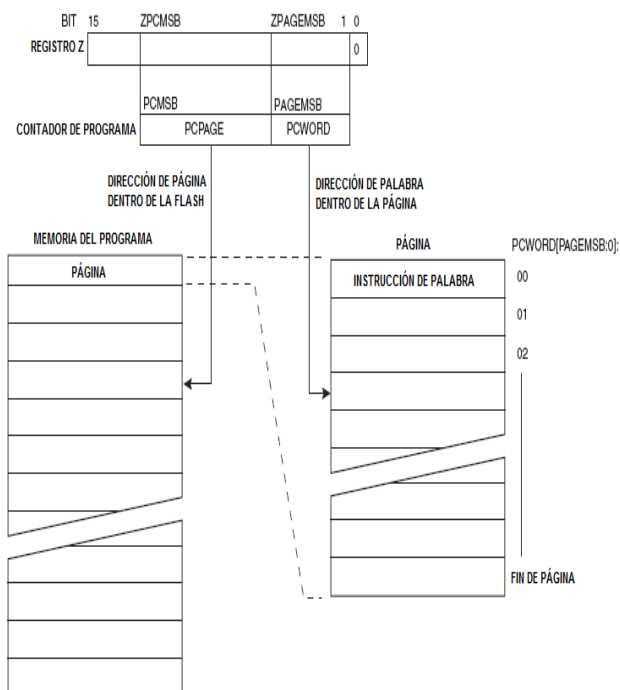


Fig. 22 Direccionamiento de la Flash durante SPM

C. Diseño del Bootloader

EL registro SPMCSR (Store Program Memory Control and Status Register) contiene los bits de control necesarios para las operaciones del BootLoader (ver figura 23).

7	6	5	4	3	2	1	0
SPMIE	RWWSB	SIGRD	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0

Fig. 23 Registro de estatus control para almacenar en la memoria del programa.

- Bit 7 – SPMIE: Habilidad de Interrupciones SPM.

- Bit 6 – RWWSB: Lectura mientras la sección de escritura esta ocupada.
- Bit 5 – SIGRD: Lectura de la Fila de Firma.
- Bit 4 – RWWSRE: Lectura-Mientras- Escribe habilita la sección de lectura.
- Bit 3 – BLBSET: Poner en uno el Bit del Boot Lock Bit Set.
- Bit 2 – PGWRT: Escritura de Página.
- Bit 1 – PGERS: Borrado de Página.
- Bit 0 – SPMEN: Habilitar la capacidad de almacenar y borrar en la memoria.

1) *Borrar por página:* El registro Z se utiliza para seleccionar la página que se va a borrar. El registro Z va a apuntar al inicio de la página para borrar de byte en byte. En el dispositivo el tamaño de página es de 64 palabras (128 bytes), los seis primeros bits del registro Z se ignoran. Para borrar una página, mediante programación los bits SPMEN y PGERS en el registro SPMCRS se activan y luego se ejecuta la instrucción de SPM.

2) *Cargando la página de Buffer:* Para escribir nuevos datos en una página de la memoria, primero se almacena en la página del buffer. Esta página se llena palabra por palabra, luego el buffer se copia en la memoria Flash en una sola operación. El registro Z se utiliza para seleccionar la palabra que se escribe en el búfer.

3) *Escribir por páginas en la memoria:* Para escribir la palabra en la memoria de la página, se carga la palabra en los registros R1 y R0. Se ajusta el registro Z para que apunte a la palabra correcta y se establece sólo el bit SPMEN en el Registro SPMCRS. La instrucción SPM a continuación se ejecutada dentro de los cuatro ciclos

Cuando el Buffer de la página se carga con nuevos datos, se escribe en la memoria Flash. Para ello en el registro Z se establece de la misma manera como se describe en la sección referente a borrado de página. A continuación se ajusta los bits SPMEN y PGWRT en el Registro SPMCR y se ejecuta la instrucción SPM dentro de los cuatro ciclos.

4) *Activación de la Sección RWW:* Al realizar un borrado de página o escritura de página en la sección de RWW, el bit RWWSB se establece por el hardware, para indicar si la sección es accesible o inaccesible. El bit RWWSB se limpia en el software cuando la operación de la SPM se ha completado

En la figura 24 Muestra el diagrama de flujo del programa principal del Bootloader, para almacenar en la memoria SRAM, borrar y escribir en la memoria del microcontrolador y la activación de nuevo de lectura de la sección RWW.

D. Ambiente Gráfico

Una vez realizado el software de Bootloader, se elaboró un ambiente visual para la descarga hacia el microcontrolador desde la PC, tomando en cuenta las características programadas en el Bootloader.

El ambiente gráfico permite que el estudiante seleccione la velocidad de transmisión, el puerto que se desea abrir, bits de paridad, bits de parada y el número de bits de información, todos estos son parámetros para la transferencia del programa de la práctica. También consta con un cuadro de diálogo para cargar el archivo de la práctica. En la figura 25 se puede ver la interfaz de gráfica desarrollada en Visual Studio 2008.

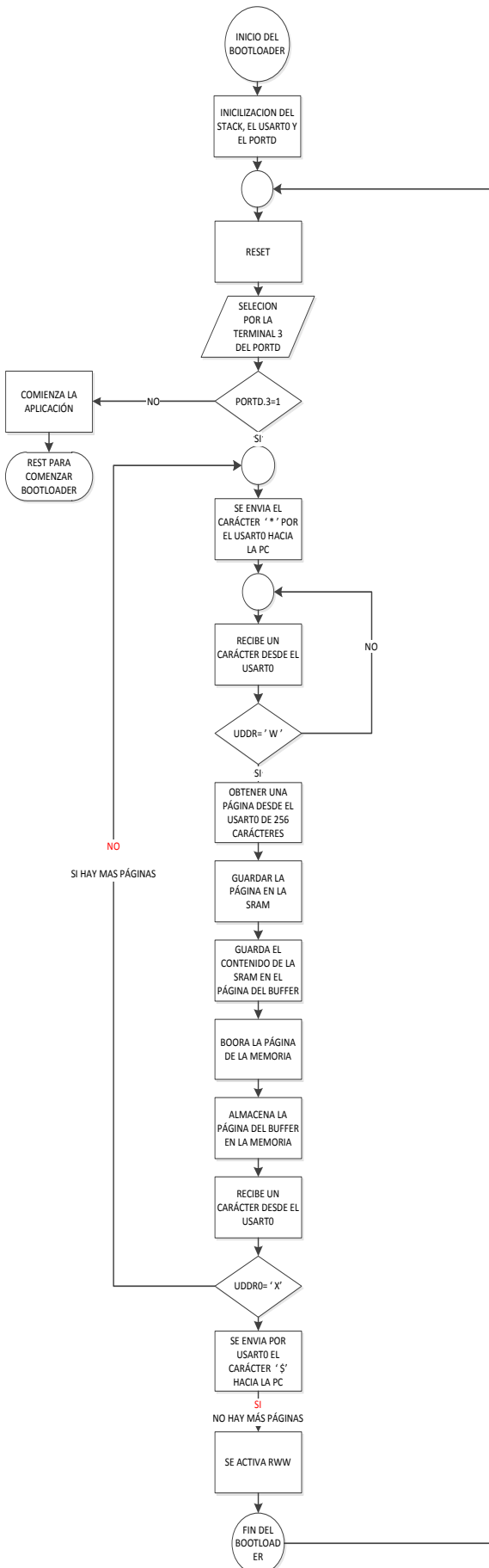


Fig. 24 Diagrama de flujo general del programa Bootloader

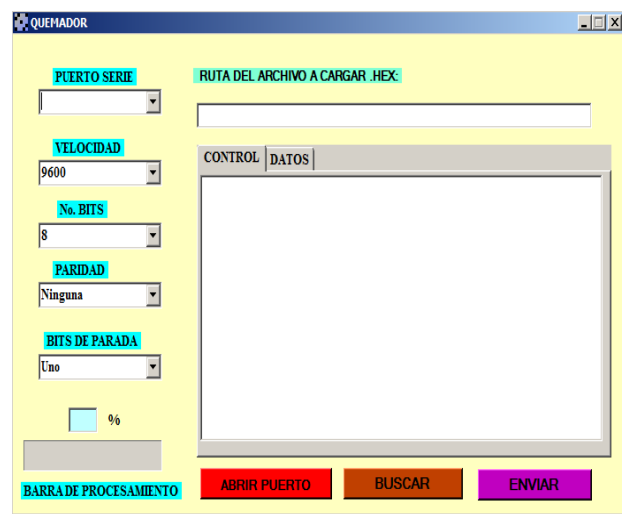


Fig. 25 Interfaz gráfica para la descarga del programa.

En la figura 28 muestra el diagrama de flujo del programa principal de la interfaz gráfica, para la descarga del código de la aplicación hacia el dispositivo.

II. EJEMPLOS DE PRÁCTICAS DE LABORATORIO

Para la realización de las prácticas se utiliza el software computacional Avr Studio 4, con su compilador WinAvr, que es compatible con ANSI C, adicionalmente se dispone de librerías específicas para el manejo de los diferentes recursos.

Se realizan las prácticas de Laboratorio de Sistemas Microprocesados, utilizando el compilador WinAvr, por lo cual se ha elaborado 3 prácticas de laboratorio, las cuales son:

- Cuenta Personas
- Cerradura Electrónica
- Voltímetro

Se va explicar la rutina para el manejo de la LCD gráfica elemento principal del dispositivo y de las prácticas.

A. Rutina para el manejo de la LCD gráfica

Primero se crea las pantallas o menús que va mostrar el GLCD, para lo cual se utiliza la herramienta Paint y el programa BMP to ASM para transformar los bits a caracteres hexadecimales, los mismos que se utilizan para mostrar en la GLCD.

Se tiene en consideración que GLCD y el teclado están conectados al mismo pòrtico, para lo cual tenemos que inicializar cada vez que vayamos usar uno u otro elemento.

Para la elaboración de una pantalla de menú primero se construye el dibujo en el software Paint para lo cual se utiliza características monocromáticas, de resolución de 128x64 pixeles en formato bitmap (.BMP).

En la figura 27 se observa la imagen que va a mostrar la GLCD creada en Paint.

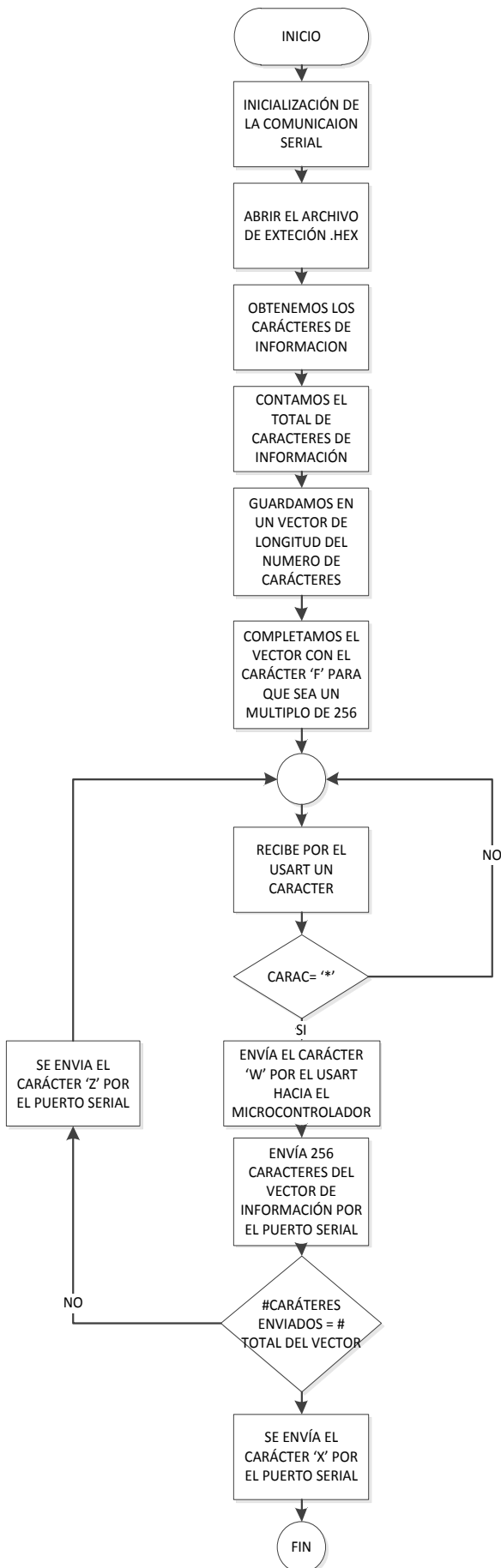


Fig. 26 Diagrama de flujo del programa principal de la interfaz gráfica

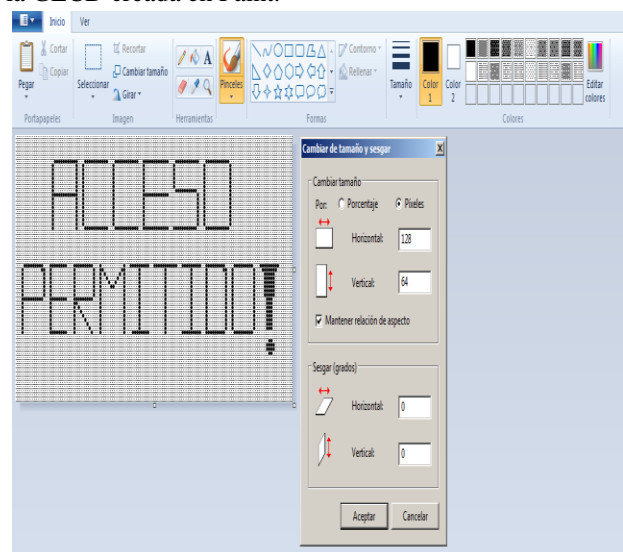


Fig. 27 Ejemplo del diseño del menú principal que va a mostrar la GLCD en el software Paint

Una vez obtenida la imagen en formato bitmap se transforma a lenguaje ensamblador utilizando un software llamado BMP to ASM, el cual se puede observar en la figura 28.



Fig. 28 Conversión de la imagen en extensión BMP a lenguaje de máquina

A continuación se muestra el diagrama de la librería que maneja la GLCD.

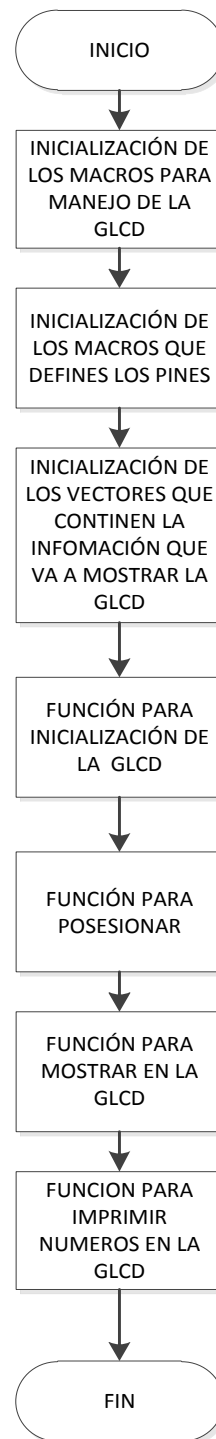


Fig. 29 Diagrama de Flujo de la Librería que maneja la GLCD

B. Práctica de cuentas personas

Esta práctica consiste en contar el número de personas que van a ingresar, salir y las que se hallan actualmente dentro de un establecimiento, esta selección se efectúa por medio del teclado; el número de personas se visualiza en la GLCD; para lo cual se utiliza la activación de las interrupciones externas INT0 e INT2.

Para poder contar las personas que ingresaron y las que salieron se usa un elemento electrónico llama optoacoplador.

En la figura 30 muestra el diagrama de flujo de la práctica de cuenta personas.

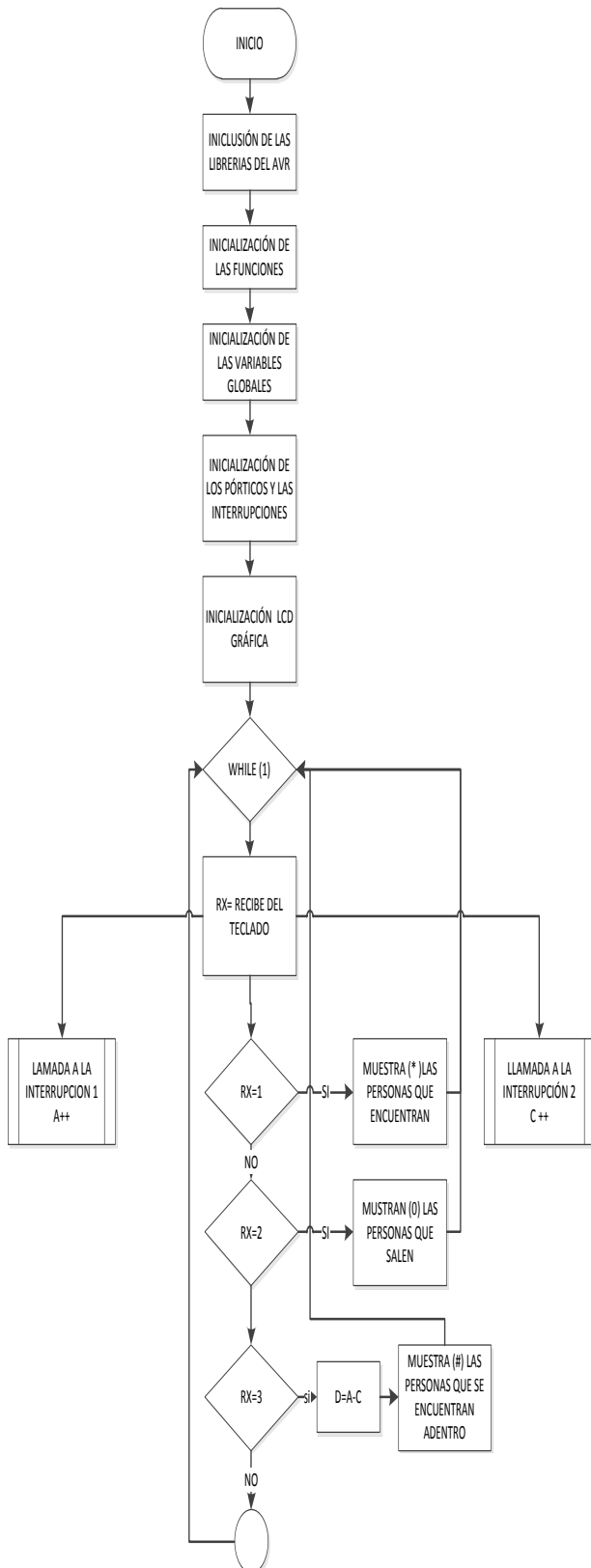


Fig. 30 Diagrama de flujo de la práctica de la cuenta personas.

C. Práctica de cerradura electrónica

En esta práctica utiliza el teclado 4x4 para introducir la clave, validarla o quitarla, se visualiza los números que

se ingresa en la GLCD, como pantallas de acceso permitido y o de acceso denegado cuando ingresemos la clave mal.

A continuación se observa el diagrama de flujo de la práctica de la cerradura electrónica.

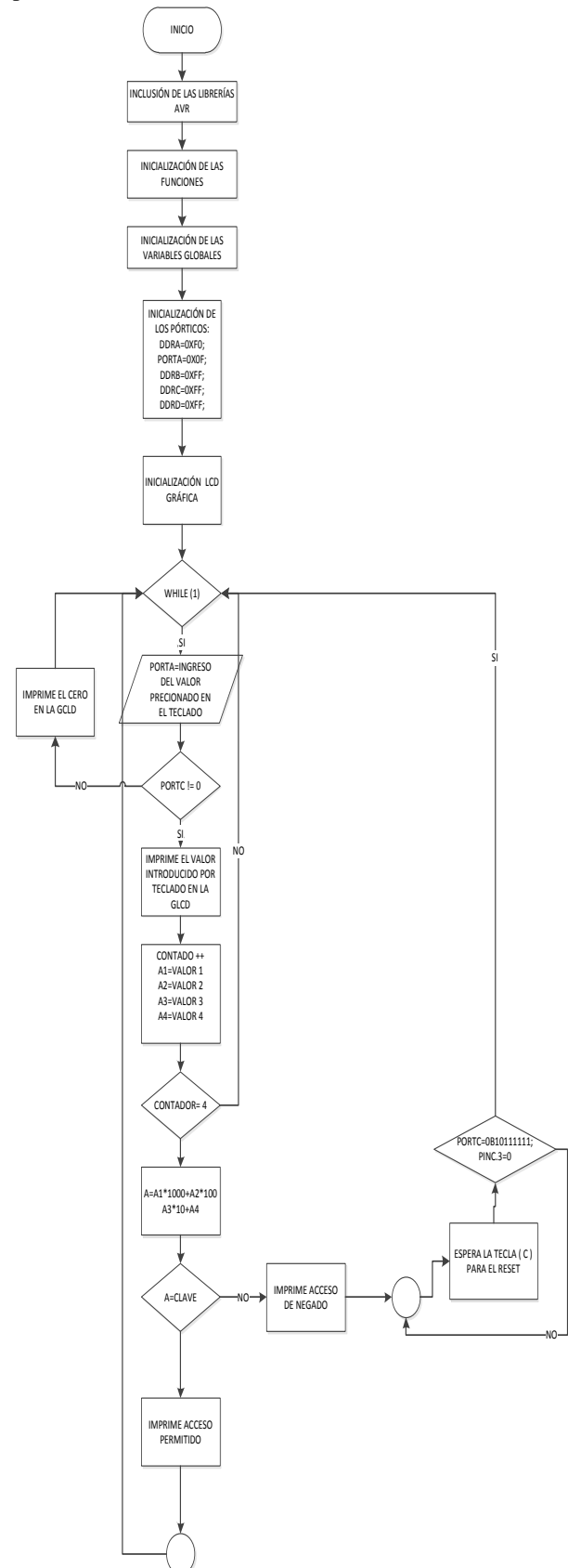


Fig. 31 Diagrama de flujo de la práctica de la cerradura electrónica.

D. Práctica del voltímetro

Se convierte la señal analógica ingresada por el PINA1 de un divisor de voltaje formado por un potenciómetro variable, que nos permitirá comprobar la conversión de la señal analógica a una señal digital del voltaje en el potenciómetro que se visualiza en la GLCD.

El diagrama de flujo de la práctica del voltímetro digital se observa a continuación.

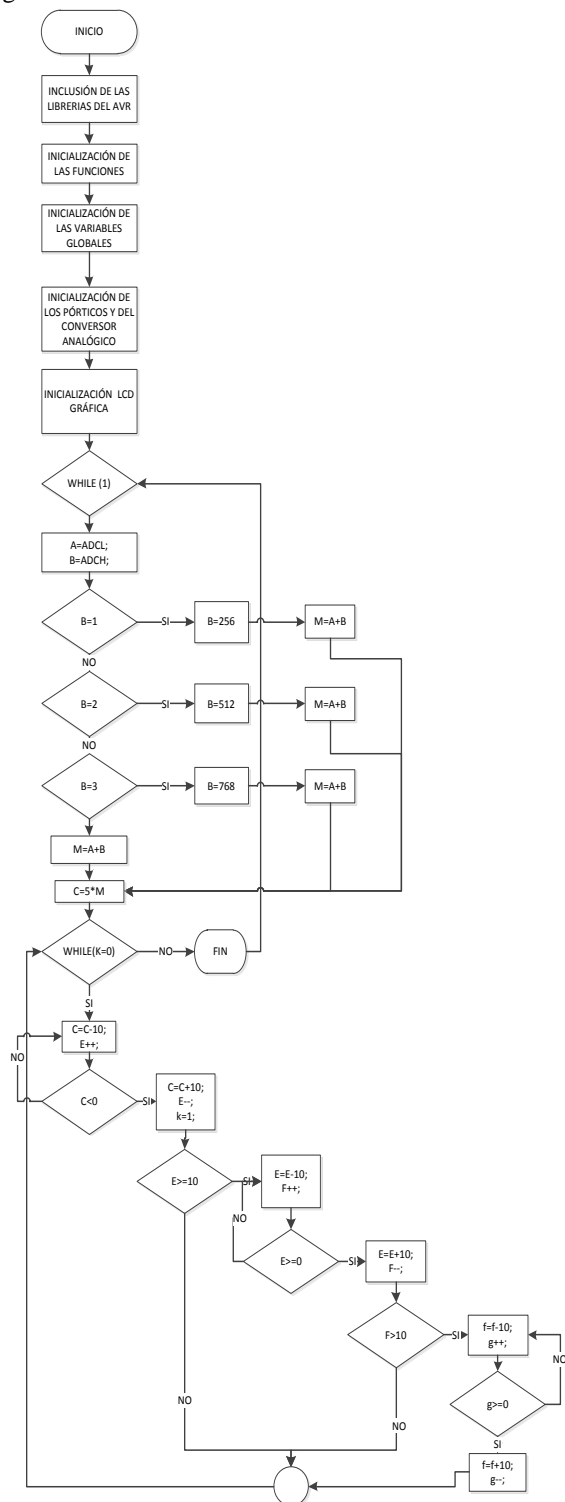


Fig. 32 Diagrama de flujo de la práctica del voltímetro.

III. PRUEBAS DE FUNCIONAMIENTO Y DESCARGA DEL PROGRAMA MEDIANTE LA COMUNICACION BLUETOOTH DEL DISPOSITIVO

El software llamado QUEMADOR se desarrollo para que el estudiante tenga una interfaz amigable para la búsqueda y envío de la práctica hacia el PDAT164P desde la computadora.

En la figura 33 se puede observar la interfaz gráfica QUEMADOR

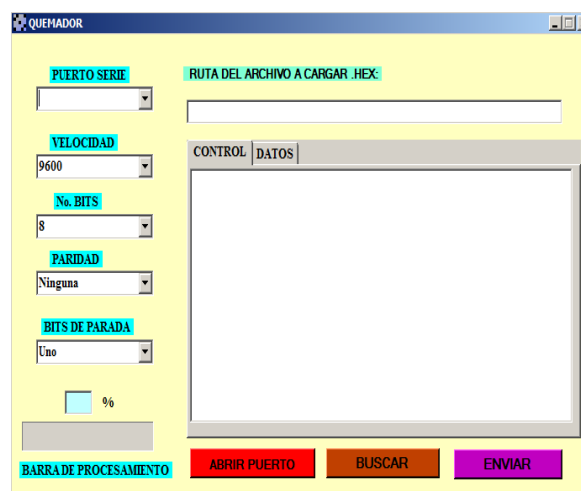


Fig. 33 Software QUEMADOR

Esta interfaz contiene las siguientes opciones:

- Parámetros de la comunicación.
- Botón de abrir el puerto.
- Botón de buscar.
- La ruta de Archivo a enviar.
- Pantallas de información.
- Botón de envío.
- Barra de procesamiento

Para poder descargar un programa primero se debe conectar el PDAT164P a la fuente de voltaje y el interruptor debe estar en la posición de Bootloader. (Ver figura 34).



Fig. 34 PDAT164P en la posición de Bootloader.

Una vez que el Prototipo esta en la posición de Bootloader se utiliza el software QUEMADOR para la descarga de la práctica. En la figura 35 se observa la finalización de la descarga del programa de la cerradura electrónica.

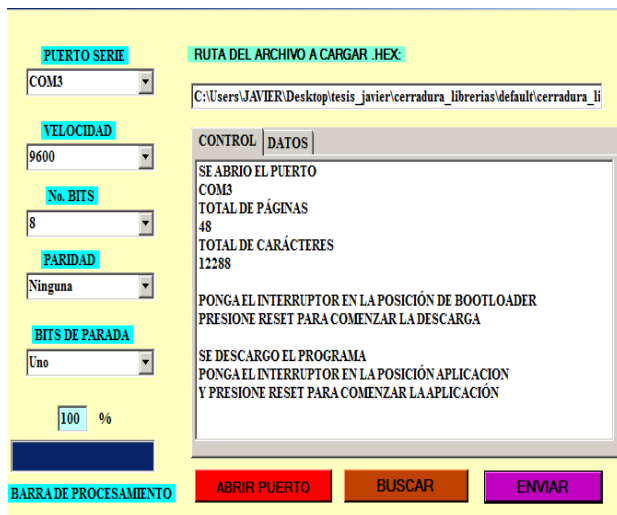


Fig. 35 Descarga de la práctica de la Cerradura electrónica

Como nos indica la pantalla de control para ir a la aplicación se debe poner el PDAT164P en la posición de “aplicación” y pulsar el reset.



Fig. 38 PDAT164P en la posición de Aplicación.

En la figura 36 se observa las características de la descarga.

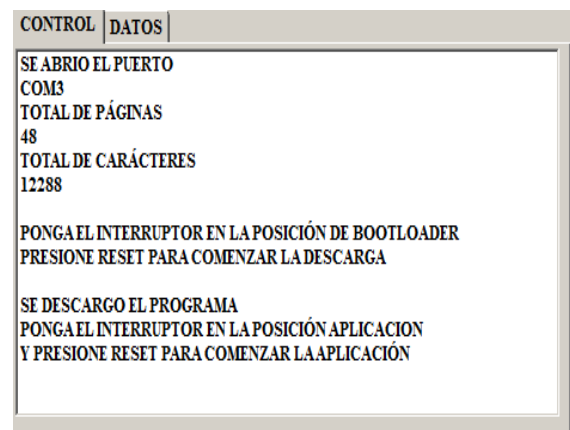


Fig. 36 Característica de la descarga del programa.

De la gráfica anterior se obtiene lo siguiente:

TABLA 2
RESULTADO DE LA DESCARGA DE LA PRÁCTICA

ENVIO	#
CARACTERES	12288
PÁGINAS	48
TAMAÑO PALABRAS/PÁGINA	64
TOTAL DE PALABRAS	3072
TOTAL DE BYTES	6144

De la tabla anterior se obtiene que se descargo mediante comunicación Bluetooth un programa de 6,144 KB a una velocidad de transmisión de 9,6 KB/s sin cometer ningún error en la descarga de la práctica, con lo que se concluye que la comunicación es eficiente debido a que el programa que se envió hacia el dispositivo es de mayor tamaño comparado con los programas que se realizan actualmente en el laboratorio que van de un rango de 1 a 700 bytes aproximadamente.

IV. CONCLUSIONES

Con la utilización del equipo el estudiante ya no necesita llevar elementos como son: microprocesador, programador, protoboard, etc., con lo cual puede enfocarse a nivel de software para desarrollar mejores sistemas inteligentes a nivel de programación. También van dirigidas a un beneficio social para los estudiantes del Laboratorio de Sistemas Microprocesados de la Escuela Politécnica Nacional, en relación al aspecto económico.

El PDAT164P tiene la posibilidad de conectarse a otros dispositivos o circuitos electrónicos por los pines libres, cuando sea el caso de obtener información de sensores u otros dispositivos electrónicos, ampliando el aprendizaje del estudiante, y así el estudio no solo se enfoca a los elementos que lo constituye.

Con el programa Gestor de Arranque que reside en la memoria de Bootloader, el microcontrolador se puede auto programar, en otras palabras el estudiante únicamente con la programación por una sola vez mediante un dispositivo ISP de este programa, ya no tiene la necesidad de utilizar otro programador, sino recurrir a la interfaz gráfica realizada en este proyecto para la descarga del programa y una comunicación serial RS-232 entre la PC y el microcontrolador.

A través de la realización de este proyecto se creó un gestor de arranque (Bootloader), el cual se utiliza para auto programar el microcontrolador, pero la utilidad del mismo tiene una variedad de aplicaciones en el mundo de la electrónica, ya que permite la actualización de los sistemas controlados por un microprocesador como por ejemplo en la Domotica o cualquier sistema inteligente; sin necesidad de desmontar ningún dispositivo, sino mediante una comunicación inalámbrica modificar los sistemas.

Para bajar el costo del equipo en vez de realizar una comunicación vía Bluetooth entre la computadora y el PDAT164P, se puede utilizar, la descarga del programa de la práctica utilizando un interfaz RS-232, obteniendo el mismo resultado.

Para la realización de las prácticas en la que se utilice la GLCD se recomienda el uso de un lenguaje de alto nivel como es C o Basic, ya que el manejo en un lenguaje como ensamblador sería tedioso y extenso para el estudiante.

REFERENCIAS

- [1] BARRETT, Steven, PACK, Daniel, "Atmel AVR Microcontroller Primer: Programming and Interfacing", Morgan & Claypool Publisher series, USA, 2008.
- [2] PARDUE, Joe, "C Programming for Microcontrollers", Editorial Smiley Micros, Knoxville, 2005
- [3] GRANIZO, Evelio, "Lenguaje C Teoría y Ejercicios", Editorial ESPE, Ecuador, 1999.
- [4] BLACO, Luis, "Programación en Visual Basic .NET", Grupos EIDOS, Madrid, 2002.
- [5] GRUNDGEIGER, Dave, "Programming Visual Basic .NET", O'Reille, USA, 2002.
- [6] http://www.atmel.com/dyn/resources/prod_documents/doc7674.pdf
- [7] Atmel, "AVR106: C functions for reading and writing to Flash memory", 2006.
http://www.atmel.com/dyn/resources/prod_documents/doc2575.pdf
- [8] Atmel, "AVR: Self Programming", 2004.
http://www.atmel.com/dyn/resources/prod_documents/doc1644.pdf
- [9] Atmel, "Efficient C Coding for AVR", 2004.
http://www.atmel.com/dyn/resources/prod_documents/doc1497.pdf
- [10] Atmel, "Atmel's Self-Programming Flash Microcontrollers", 2003.
http://www.atmel.com/dyn/resources/prod_documents/doc2464.pdf
- [11] Rovinc Networks "Hoja de datos de RN-41", 2009
<http://www.rovincnetworks.com/documents/RN-41.pdf>
- [12] Visual-paradigm, "Programin in VB.NET"
http://media.visual-paradigm.com/media/documents/dbva40npg/pdf/dbva_dotnet_programmer_guide_chapter7.pdf



Ing. Javier De la Torre G.

Nació en Quito-Ecuador el 12 de noviembre de 1986. Graduado en Ingeniería en Electrónica y Telecomunicaciones en el 2011 en la Facultad de Ingeniería Eléctrica y Electrónica de la Escuela Politécnica Nacional (EPN). Actualmente se desenvuelve como Especialista en Telecomunicaciones en SONDA S.A.

Ing. Jaime E. Velarde.

Graduado en Electrónica y Telecomunicaciones - 1980 EPN. Profesor del DETRI desde 1977. Participante por la EPN en la Red ALFA: "Creación de una biblioteca de Objetos de Aprendizaje, orientados a la formación en las TICs". Administrador del primer Campus Virtual de la EPN para la capacitación de los docentes con FATLA "Fundación para la aplicación de la tecnología en Latinoamérica". Participante en el Proyecto de Investigación: SIMON-VE "Sistema Integral de Monitoreo de Vehículos". jaime.velarde@epn.edu.ec