IMPLEMENTACIÓN DE UN MICROPROCESADOR UTILIZANDO UN ATMEGA 8515.

Presentado como requisito de la asignatura Microcontrolaldores, dictada en la universidad del Quindío durante el primer semestre del año 2016.

Presentado por:

Santiago Restrepo Ruiz

Ronald Andrés Mora

Nicolás Gutierrez Gonzalez

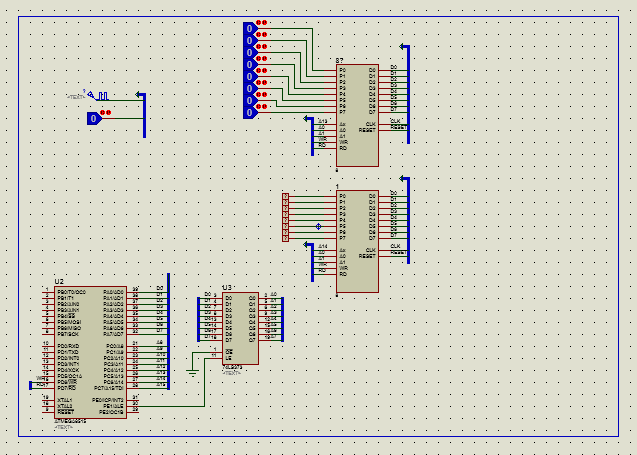
Presentado a:

Luis Miguel Capacho V.

Gerardo Andrés López O.

Armenia, Quindío en Colombia.

Mes de Marzo de 2016.



*Figura 1, Circuito implementado hasta el momento.*

Un microcontrolador es un pequeño ordenador que contiene en su interior básicamente un procesador, soporte (reloj y reset), memoria y puertos de entrada-salida, hasta el momento el sistema no cuenta con memoria; Posee dos puertos, cada uno con 8 pines, por ahora un puerto está dispuesto como un elemento de recolección de datos y otro, dispone sus 8 pines para mostrar lo que se lee en el primero. Esta operación la realiza el microprocesador, que para este caso es un ATEMEGA 8515, que funciona siguiendo un algoritmo codificado en lenguaje C utilizando la plataforma Codebloks.

El verdadero propósito de este informe es mostrar el funcionamiento del código implementado para el funcionamiento del ATMEGA. Código que aparece a continuación:

#include "procesador.h"

int main(){

uint8\_t value;

CPUInit();

CPUWrite(0x2000, 0xFF);

CPUWrite(0x4000, 0x00);

while(1){

value = CPURead(0x2002);

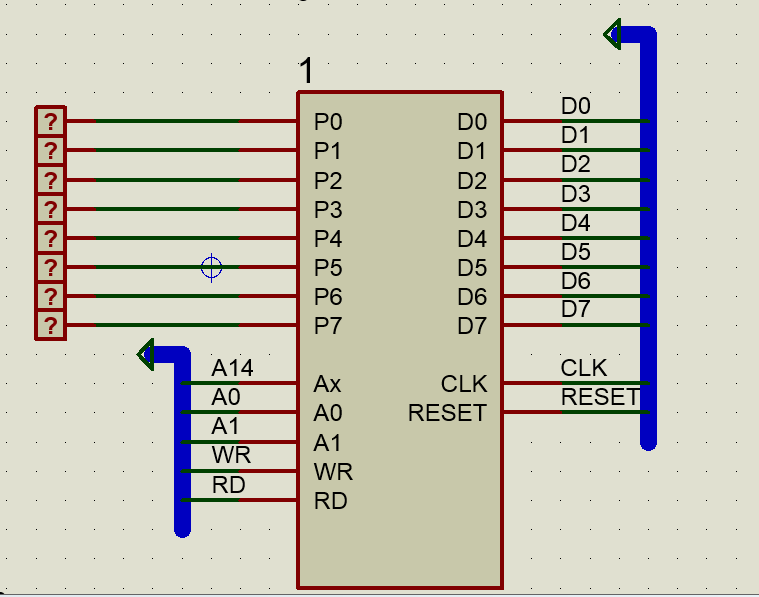
CPUWrite(0x4001, value);

}

*Código 1. Configuración de lectura y escritura.*

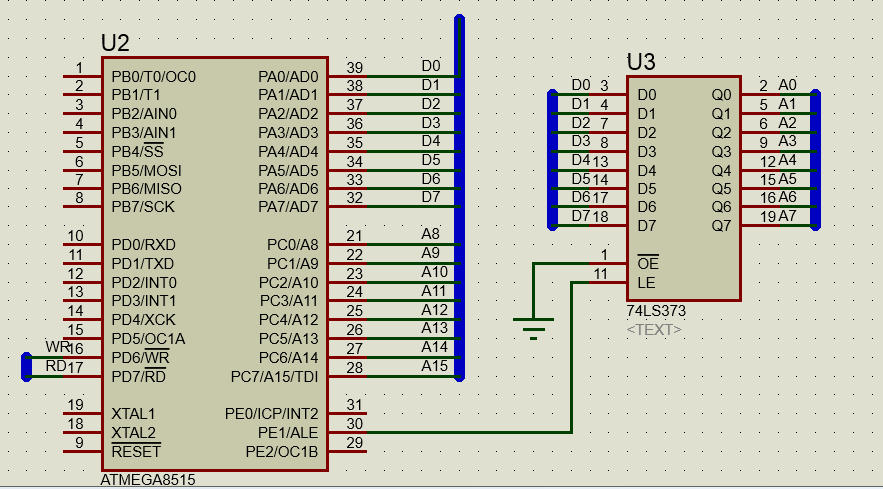
El código anterior es tal que dispone el sistema con la ayuda de un par de bibliotecas, para comprobar en tiempo real el funcionamiento del programa y detectar errores en el mismo de manera sencilla, a tal mecanismo se le conoce como programa de depuración o monitorización (debugger).

La función CPUWrite, recibe la dirección del puerto, y el valor de voltaje en hexadecimal, y activa la funcionalidad WR, tal que 0xFF son 5 voltios y 0x00 son cero, esta es la función que es capaz de modificar el comportamiento del mismo; La función CPURead, recibe la dirección del puerto, y activa la funcionalidad RD, esta permite conocer datos en el puerto. la dirección 0x2000, modifica la funcionalidad del puerto, 0xFF configura el mismo como entrada, 0x00 lo dispone para leer valores, esto si además se activa la funcionalidad WR. La dirección 0x2001 permite ingresar un valor para ser indicado a la salida de un pin en un puerto, si se activa la función WR y permite conocer el valor a la salida si además de esta dirección se activa la función RD. La dirección 0x2002 permite conocer el valor que hay en cualquier pin de ese puerto, sí previamente se ha configurado el mismo como de lectura (acá debe estar activada la funcionalidad RD).



*Figura 2, esquema de un Puerto configurado como salida.*

**Control de puertos:** La figura en la parte superior muestra uno de los puertos, específicamente aquel que está configurado como salida, los pines marcados con el índice P y el subíndice numérico representan las salidas de cada uno de los 8 pines de éste puerto. Los índices D, A y subíndice número, indican las direcciones, WR y RD indican la disposición de lectura o escritura.



*Figura 3, ATMEGA 8515.*

En la figura 3 está el esquema del procesador, al que se le ha ingresado el algoritmo visto anteriormente (véase código 1) este procesa la información leída en un puerto y ejecuta una función que le permite mostrar la información en otro, el componente a su derecha (74LS373) está ahí para permitir al procesador distinguir entre datos y señales; cuándo la línea LE está activa, entonces el procesador puede hacer uso de las direcciones A0, hasta A15, cuándo la línea LE muestra un cero lógico, entonces, el procesador puede sólo hacer uso de las direcciones A8 hasta A15, las líneas que estaban dispuestas para las primeras 8 direcciones, son ahora los canales por los que se envían los datos. Es activando LE que el procesador elije si desea enviar datos o direcciones a los puertos o a cualquier otro elemento del microcontrolador.