

Proyecto 1- Microcontrolador con CPU AVR

Andrés Felipe Sánchez Sánchez, Nicolás Arias Mendoza.

Ingeniería Electrónica

Universidad del Quindío

afsanchez@uqvirtual.edu.co, narias@uqvirtual.edu.co

Resumen- Los microprocesadores son circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica.

En este informe se podrá ver el procedimiento realizado para lograr la implementación simulada de un Microcontrolador en la herramienta computacional de PROTEUS, se presentaran ayudas visuales de los diferentes componentes, desde la creación de los pines de entrada/salida, pasando por la implementación de la RAM y finalmente del Temporizador, para posteriormente visualizar el funcionamiento del Microcontrolador con todos sus componentes.

Palabras Clave- Microcontrolador, RAM, pines, temporizador.

I. INTRODUCCIÓN

La microcontroladores son circuitos integrados programables, capaz de ejecutar las órdenes grabadas en su memoria, actualmente en el mercado se pueden encontrar microprocesadores de 8, 16, 32, 64 bits. Para este informe se dará a conocer un procedimiento para implementar, mediante simulaciones de Proteus, un Microcontrolador de Atmel8515 de 8 bits

II. MARCO TEÓRICO

Microcontrolador:

Es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica.

RAM:

La memoria RAM es utilizada como memoria de trabajo de computadores para el sistema operativo, los programas y la mayor parte del software.

TIMER:

Es el encargado de controlar la señal de salida de un PWM.

TCCRn:

Este es el de control del temporizador, el cual permite modificar el funcionamiento del temporizador.

OCRn:

Este es el registro de datos usado para almacenar un valor de comparación. El valor almacenado en este registro es comparado con el valor actual del temporizador para generar interrupciones o modificar el ancho de pulso de una señal PWM.

TCNTn:

Este es Registro de datos que contiene la cuenta actual del temporizador.

III. PROCEDIMIENTO

Para la implementación de un solo pin, el docente Luis Miguel Capacho, facilitó un circuito, el cual después de realizar la explicación del mismo, se procedió a diseñar en la herramienta computacional de ISIS PROTEUS. A continuación se mostrara como primera medida el circuito implementado en proteus (Figura 1) y luego se procederá a explicar el funcionamiento más detalladamente.

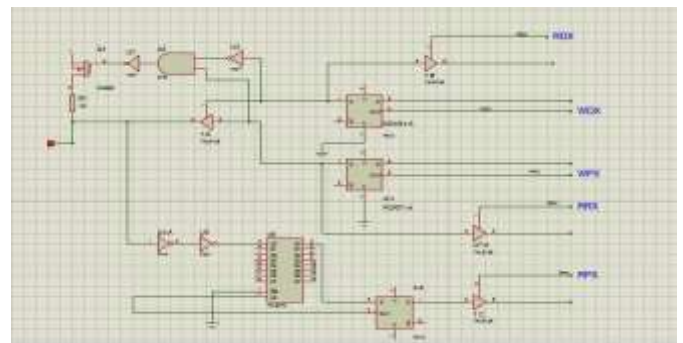


Figura 1. Circuito para un solo PIN

En la figura 1 se pueden encontrar los terminales al aire, ya que se realizó solo para mostrar la propuesta del diseño, a continuación en la figura 2, se pueden observar probadores lógicos los cuales permitirán realizar las correspondientes pruebas para determinar si el circuito está funcionando; al clonar el repositorio se podrá observar que los circuitos si se encuentran funcionando y que la prueba se realizó satisfactoriamente.

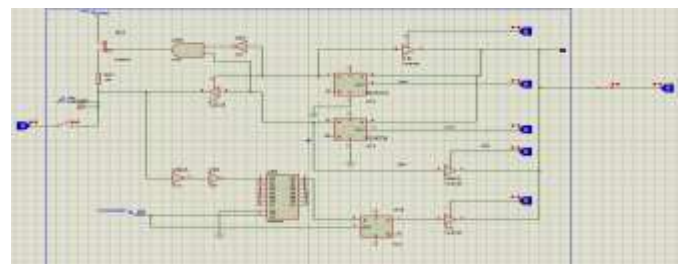


Figura 2. Circuito con probadores lógicos.

Luego en la figura 3 se puede ver la implementación de un componente electrónico en proteus, en el cual iría el circuito ya mencionado anteriormente; en la imagen se puede observar el circuito a la derecha mientras que a la izquierda se puede ver el prototipo final de la pastilla. Para esta pastilla también fue necesaria la realización de pruebas, dichas pruebas también fueron realizadas con puntas lógicas, en la figura 4, se puede observar solo la simulación de la pastilla.

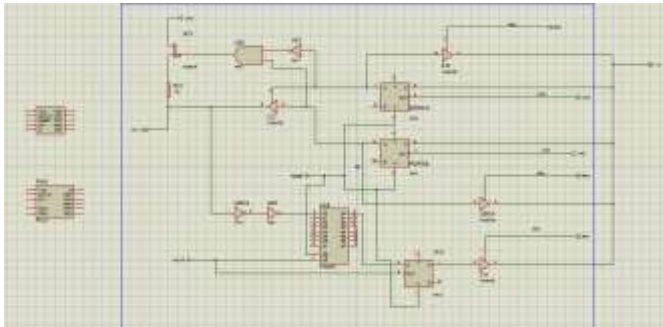


Figura 3. Implementación de la pastilla de un solo pin.

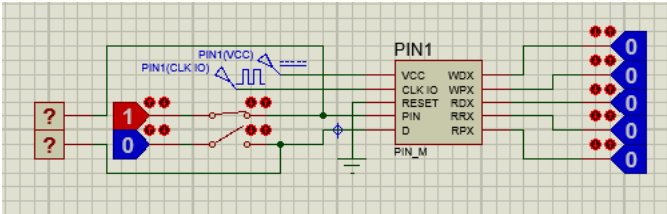


Figura 4. Prueba realizada al nuevo componente.

Para las pruebas realizadas en los circuitos de las figuras 2 y 4, fue utilizada una tabla (ver figura 5) presente en el datasheet del Micro controlador Atmel8515.

DDxn	PORTxn	PUD (in SFIOR)	I/O	Pull-up
0	0	X	Input	No
0	1	0	Input	Yes
0	1	1	Input	No
1	0	X	Output	No
1	1	X	Output	No

Figura 5. Tabla datasheet comprobación de los circuitos.

El proceso que siguió la implementación y prueba del nuevo componente, consistió en evidenciar el funcionamiento de ocho pines, los cuales estarán unidos a circuitos como compuertas lógicas que pretenden simular las referencias RDX, WDX, WPX, RRX y RPX, en la figura 6, se pueden observar la implementación de dichas compuertas.

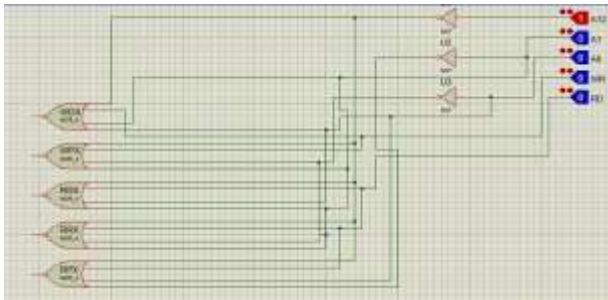


Figura 6. Compuertas lógicas simulando las referencias RDX, WDX, WPX, RRX y RPX.

La prueba de los ocho pines se puede verificar en el repositorio, la figura 7 muestra el circuito completo para lograr verlo de una forma general, con el fin de conformar la pastilla con los ocho pines ya como componente único, como la parte superior derecha de la figura 8 hace referencia la imagen anterior, se omitirá para poder realizar un Zoom al circuito del lado derecho figura 9, con el fin de visualizar con mayor claridad la implementación.

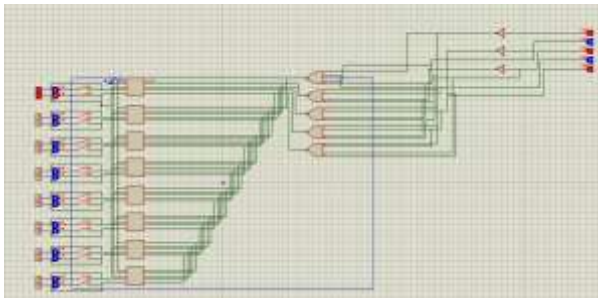


Figura 8. Circuito general implementado para los ocho pines.

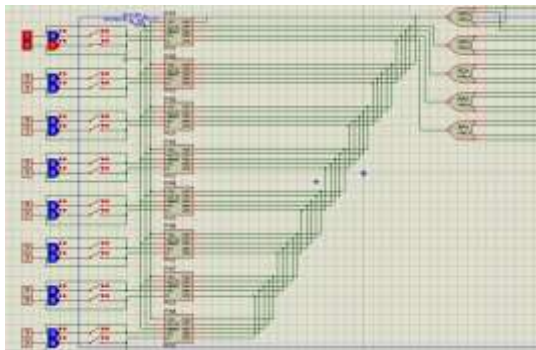


Figura 9. Circuito implementado para los ocho pines omitiendo las compuertas lógicas.

Para el proceso final de los pines, se encapsularon todos los componentes de la figura anterior en un solo y completo componente, ver figura 10.

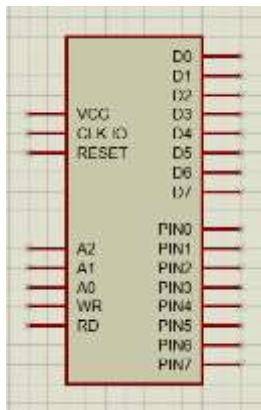


Figura 10. Componente de los ocho pines.

En la figura anterior se puede observar el componente de los ocho pines, pero la prueba sería realizada con el procesador ATMEGA8515, dicho procesador es una pastilla ya incluida en los componentes de proteus. Para esta prueba fue necesario copiar unos archivos de códigos para CodeBlocks suministrados por el docente Luis Miguel Capacho, los cuales tenían ya un programa, mediante el cual sería permitida la prueba de el nuevo componente, la prueba con el microcontrolador puede ser observada en la Figura 11.

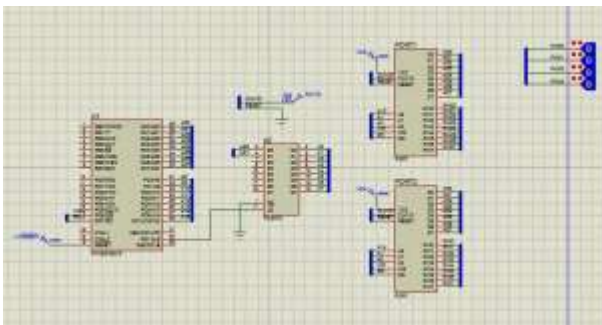


Figura 11. Prueba componente con micro-procesador ATMEGA8515

A continuación se puede observar, Figura 12, el montaje realizado para la implementación del circuito completo con la RAM.

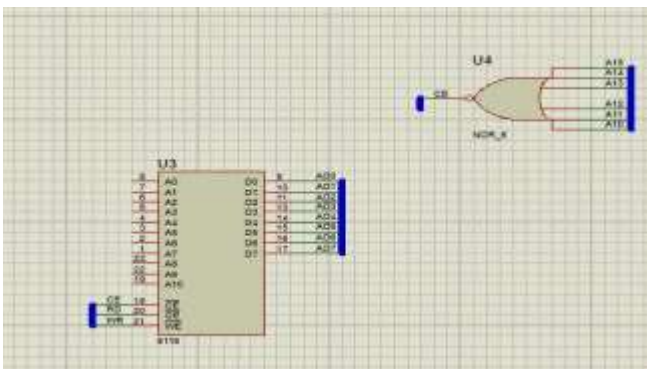


Figura 12. Circuito de la RAM

Cuando ya se realizaron las pruebas de la RAM, se procedió a realizar la implementación del TIMER (Temporizador de 8 bits), para esto fue necesario como ayuda un documento enviado por el docente Luis Miguel Capacho.

Para esta parte, se implementaron unos circuitos, los cuales se pueden ver representados en la figura 13, desde este esquema se fueron analizando parte por parte y realizando la implementación de cada uno de los circuitos.

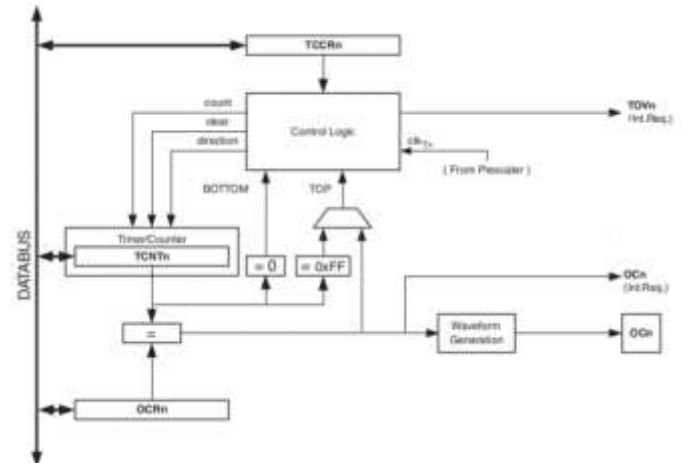


Figura 13. Temporizador 8 Bits

La figura 14, muestra el registro TCCRn, este es el de control del temporizador, el cual permite modificar el funcionamiento del temporizador.

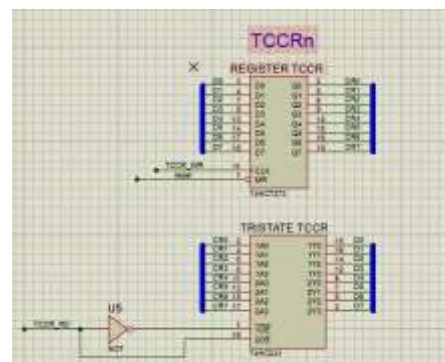


Figura 14. Circuito del registro TCCRn

El segundo circuito (figura 15) hace referencia a las siglas OCRn, este es el registro de datos usado para almacenar un valor de comparación. El valor almacenado en este registro es comparado con el valor actual del temporizador para generar interrupciones o modificar el ancho de pulso de una señal PWM.

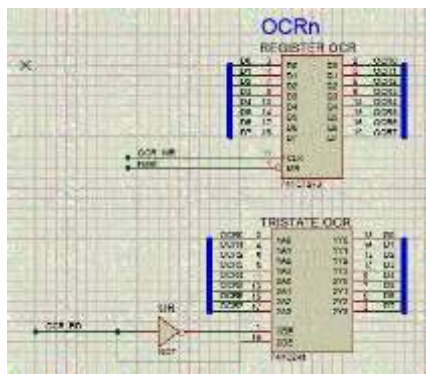


Figura 15. Circuito para el registro OCRn

Por último, el circuito de la figura 16, hace referencia a las siglas TCNTn, este es Registro de datos que contiene la cuenta actual del temporizador.

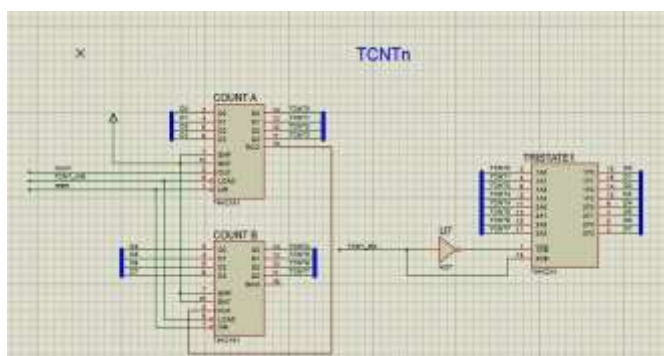


Figura 16. Circuito implementado para el registro TCNTn

Posteriormente fue necesaria la implementación de otros circuitos, tales como control lógico, generador de señal, contador, pre escalador entre otros que serán mostrados en el transcurso de esta sección, para al final integrarlo todo en una pastilla y poder obtener ya el TIMER.

El circuito que se puede observar en la figura 17, hace alusión al Control Lógico del montaje general.

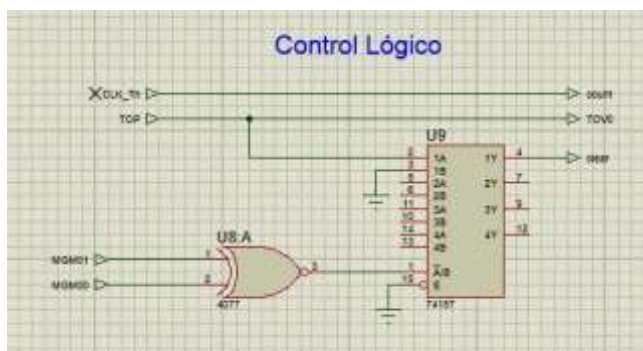


Figura 17. Circuito del control lógico.

En la figura 18 se puede visualizar el circuito desarrollado para obtener el generador de señales que posteriormente sería adherido a otro circuito más completo para implementar el circuito de salida.

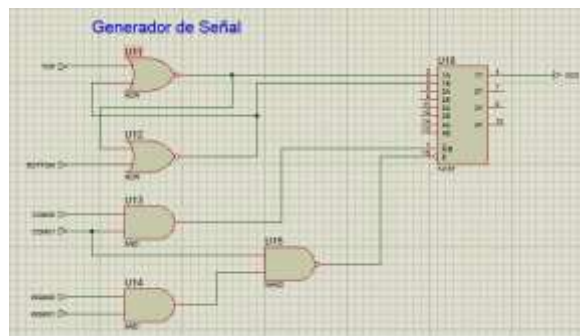


Figura 18. Circuito del Generador de señal

Para obtener el pre escalador, primero fue necesaria la implementación de un contador de 8 bits, y cuando dicho circuito estuviera listo poder adherirlo a un multiplexor para así completar el circuito del Pre escalador.

En la figura 19 se puede observar el contador implementado y posteriormente en la figura 20 se puede apreciar el pre escalador con su instalación ya completa, en la figura 19 se observa que las salidas del contador están unidas a un bus de datos, pero a continuación en la figura 20, se puede ver el pre escalador completo.

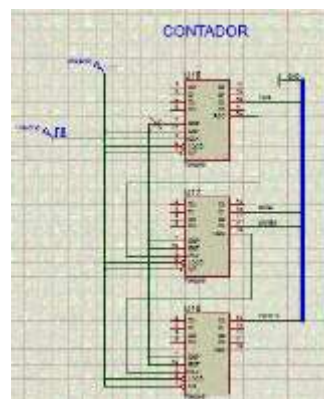


Figura 19. Circuito del contador.

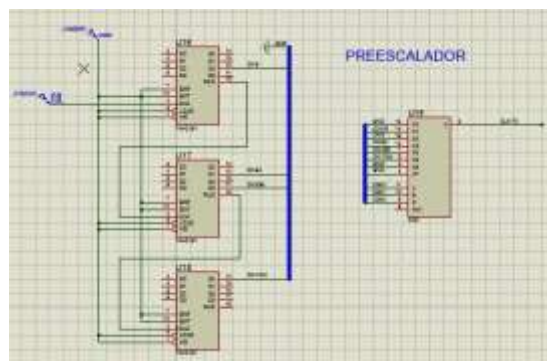


Figura 20. Pre-escalador completo

Por último se implementaron tres comparadores, el comparador =, comparador =0 y el comparador= 0xFF, estos circuitos se pueden observar en las figuras 21,22 y 23 respectivamente.

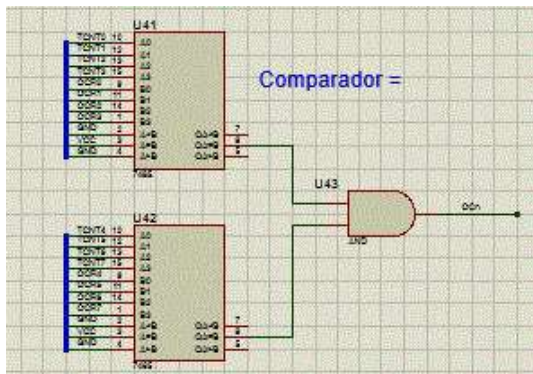


Figura 21. Comparador =.

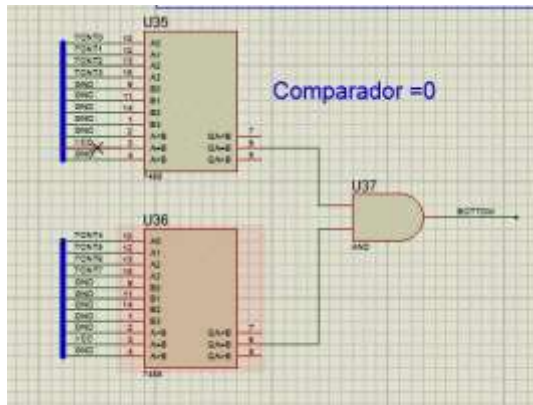


Figura 22. Comparador =0.

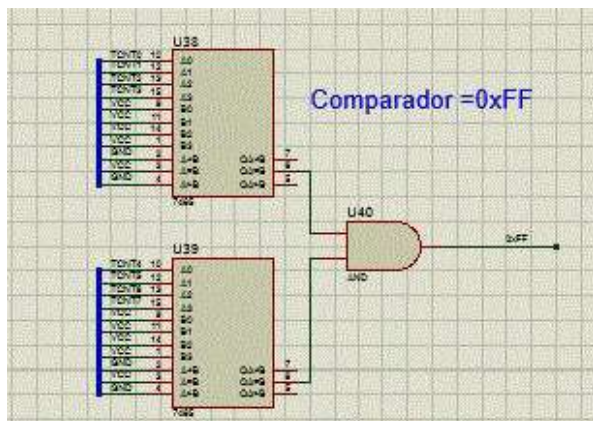


Figura 23. Comparador =0xFF.

Como se puede ver en la figura 13, aún faltan hacer los circuitos correspondientes a un multiplexor entre las salidas de los comparadores y obteniendo como resultado el registro TOP (Ver figura 24), para el generador de señal, figura 18, se le agrego un nuevo circuito correspondiente al de salida para el PWM por lo tanto en la figura 25 se puede observar el generador con la salida a la derecha de la imagen.

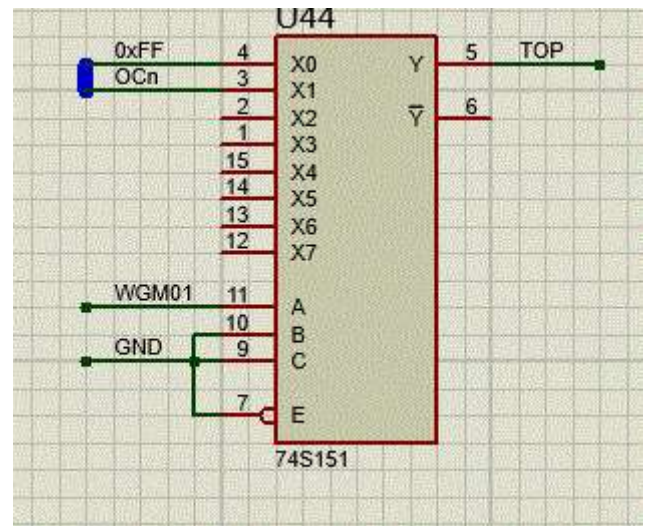


Figura 24. Multiplexor.

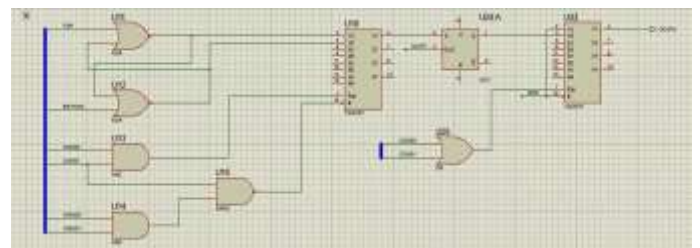


Figura 25. Generador de señal con circuito de salida.

Por último, las figuras 14,15 y 16 hacen referencia a los registros TCCRxn, OCRxn y TCNTxn respectivamente. En estos circuitos se puede observar que existen entradas a los mismos las cuales no se saben cuáles son, en las figuras 26,27 y 28 se puede observar que cada uno de esos datos que ingresan a cada pastilla corresponde a un circuito a parte, los cuales se mostraran a continuación.

El registro TCCR, figura 26, son valores de entrada para el registro TCCRxn.

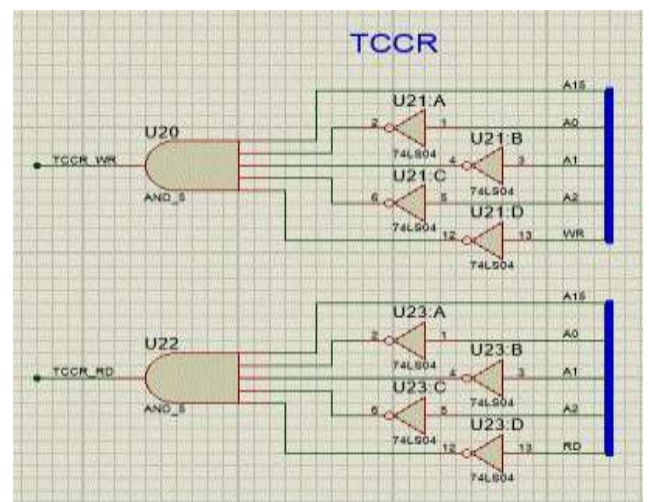


Figura 26. Registro de ingreso a TCCRxn

Para el registro OCRxn están los siguientes circuitos figura 27, haciendo referencia a dos valores de entrada.

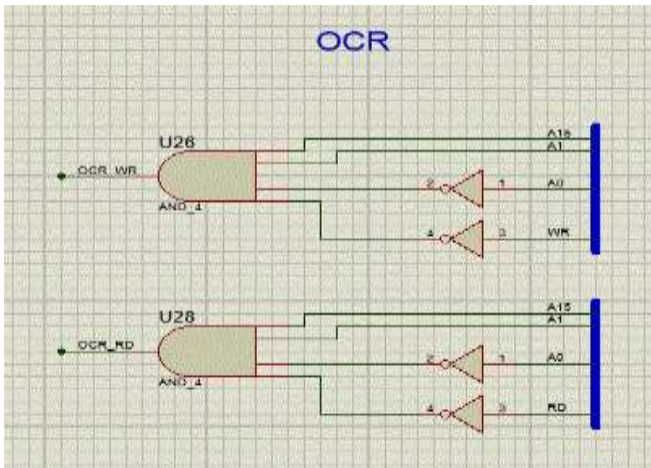


Figura 27. Datos de entrada para el registro OCRxn.

Y para el último registro TCNTxn los datos que tiene como entrada sería el registro TCNT que se puede observar en la figura 28.

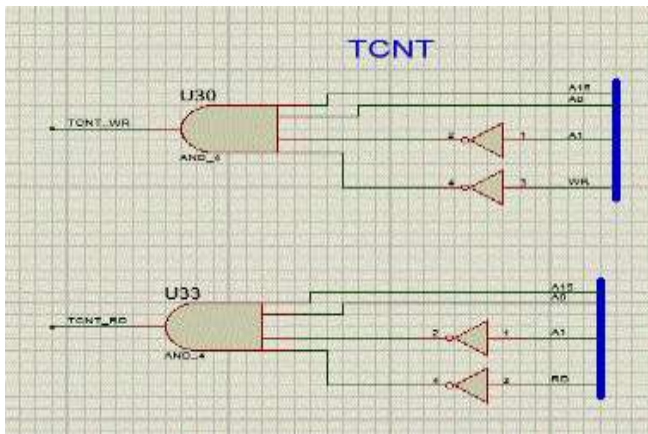


Figura 28. Registros de ingreso a TCNTxn.

Ya con todos estos circuitos unidos se tiene como resultado el Temporizador y posterior mente se procederá a agregarlo a los circuitos ya implementados con anterioridad para completar el Microcontrolador.

El código utilizado para realizar la prueba del temporizador es el siguiente:

```
#include "procesador.h"
#include "map.h"
int main()
{
    uint8_t pulse;
    CPUInit();
    CPUWrite(DDR_A, 0); //Se define el puerto A como
    entrada 0x2000
    CPUWrite(DDR_B, 0xFF); //Se define el puerto B
    como salida 0x4000
```

```
CPUWrite(TCCR_0, 0b01101101); //Se configura el
TIMER en modo de operación PWM con preescalado
de 1024
```

```
while(1){
    pulse = CPURead(PIN_A); //Se lee la dirección
    0x2002
```

```
CPUWrite(OCR_0, pulse); //Se da un impulso en
el registro OCR_0
```

```
CPUWrite(PORT_B, CPURead(TCNT_0)); //Se
escribe en la posición 0x2001 la lectura de TCNT_0
    _delay_ms(30); //Pausa de 30ms entre cada
    iteración
```

```
}
```

```
}
```

NOTA: Todos las pruebas implementadas en la realización de este informe, pueden ser corroboradas mediante la clonación y compilación de los mismos circuitos subidos en el repositorio de nuestro grupo, después de clonar el repositorio copiar los archivos contenidos en la carpeta Recursos en la carpeta MODELS de Proteus.

IV. CONCLUSIONES

Los microprocesadores actualmente soy muy importantes en nuestro entorno, ya que están presentes en casi todas las herramientas tecnológicas que utilizamos. Por eso se realizó este proyecto, para comprobar el funcionamiento de circuitos tan pequeños y a partir de decisiones simples se puedan crear sistemas más complejos con muchas funciones extras.

También se corroboro mediante el análisis y la lógica, la creación de sistemas simples que cumplan las características, que al aumentar la capacidad va dando robustez al sistema, haciendo que tenga muchas funcionalidades, a partir de arquitecturas RISC que fue la que usamos

V. BIBLIOGRAFIA

- Datasheet Microcontrolador Atmel 8515
- Guías suministradas por el docente Luis Miguel Capacho