

*Programando Embedded System Microcontroller con Metología interactiva.*

**GILBERTO PASCUAL, 2019**  
[gpascualt@gmail.com](mailto:gpascualt@gmail.com)

# Passive Avr Programmer



Figura1



Figura2

*“Ahora usted puede aprender a Programar Microcontroladores de forma muy fácil”.*

## Introducción:

El siguiente proyecto se trata de utilizar las funciones básicas para poder encender y apagar un Led con una frecuencia de 1Hz.

## Esquema electrónico:

El siguiente esquema electrónico muestra la implementación de este proyecto en un ambiente de test o simulación, mediante el cual podemos ver el funcionamiento del software e interacción con el hardware:

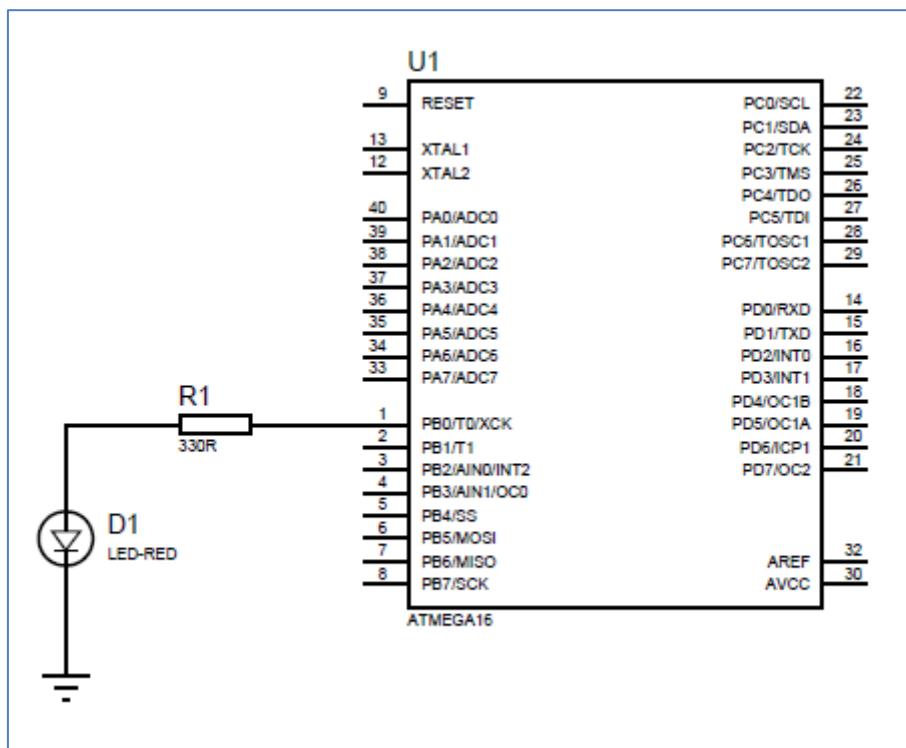


Figura3

**Nota:** El esquema presentado no es que necesariamente el que se implementaría en un ambiente real.

## Programa de Control [Firmware]:

En la Figura5, se puede apreciar el código escrito en leguaje Basic para controlar el encendido y apagado de un Led cada 1seg, recordar la formula mostrada en la Figura4:

$$T = \frac{1}{f} = 1\text{seg} \Rightarrow f = 1\text{hz}$$

Figura4

```
$regfile = "m8def.dat"
$crystal = 8000000
Config Portb.0 = Output

Led Alias Portb.0
Led = 0

Do
  Toggle Led
  Wait 1
Loop

End
```

Figura5

## Análisis del código del Programa:

1. **\$regfile = "m8def.dat"** Declaramos el Microcontrolador que usaremos, con esto el compilador conocerá que componentes internos podemos usar.
2. **\$crystal = 8000000** Declaramos la frecuencia con la cual trabajara nuestro Microcontrolador, puede usarse un cristal interno o externo.
3. **Config Portb.0 = Output** Configuramos el puerto B0, del Microcontrolador como salida.
4. **Led Alias Portb.0** Renombramos el código que llama al pin B0, con la palabra “Led”.

5. **Led = 0** Inicializamos el pin B0 con “0”, para empiece apagado.
6. **Do** Creamos una sentencia secuencial, mediante “Do”, hacemos repetitivo todo el código que se encuentra dentro de lazo “Do” y “Loop”.
7. **Toggle** Led Invierte el bit o realiza el complemento del bit, es decir si es “1”, al aplicar el comando “toggle” lo convierte en “0”.
8. **Wait 1** Con esta palabra hacemos una espera de 1 segundo, si deseamos tiempos de milisegundos debemos usar el comando “Waitms”.
9. **Loop** Para cerrar la secuencia condicional “Do”.
10. **End** Finalizamos el programa y damos fin del código

## Simulación del proyecto:

En la Figura6, se puede apreciar en el momento el cual se encuentra apagado el Led:

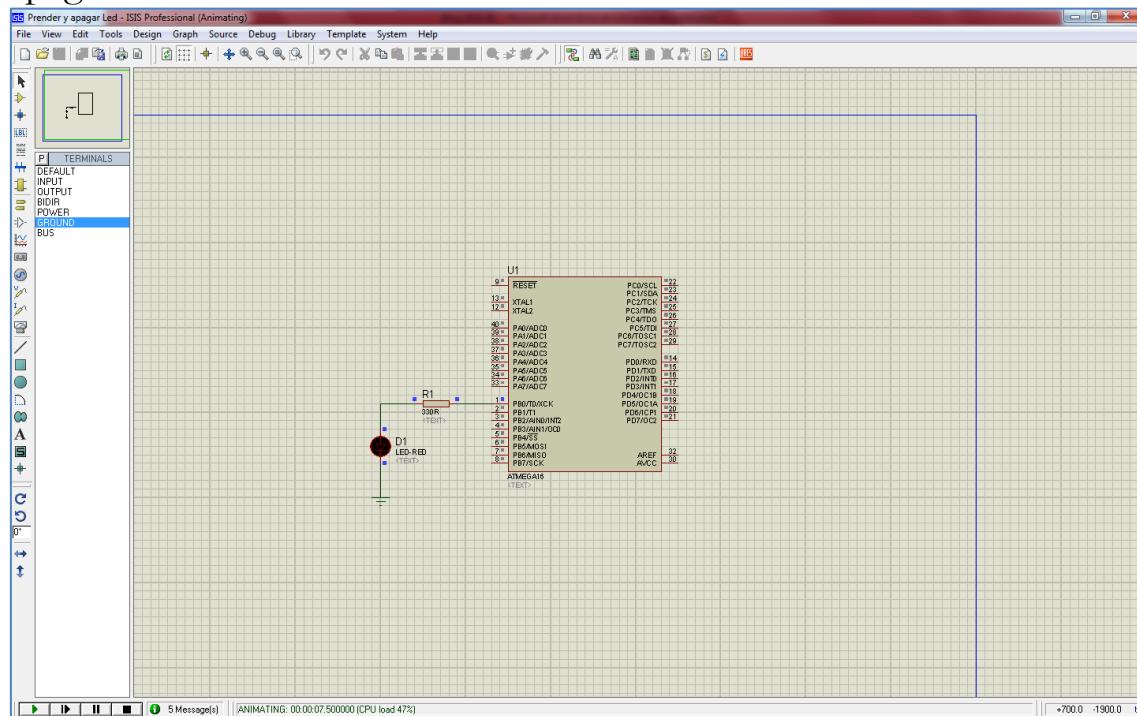


Figura 6

En la Figura 7, se puede apreciar en el momento el cual se encuentra prendido el Led:

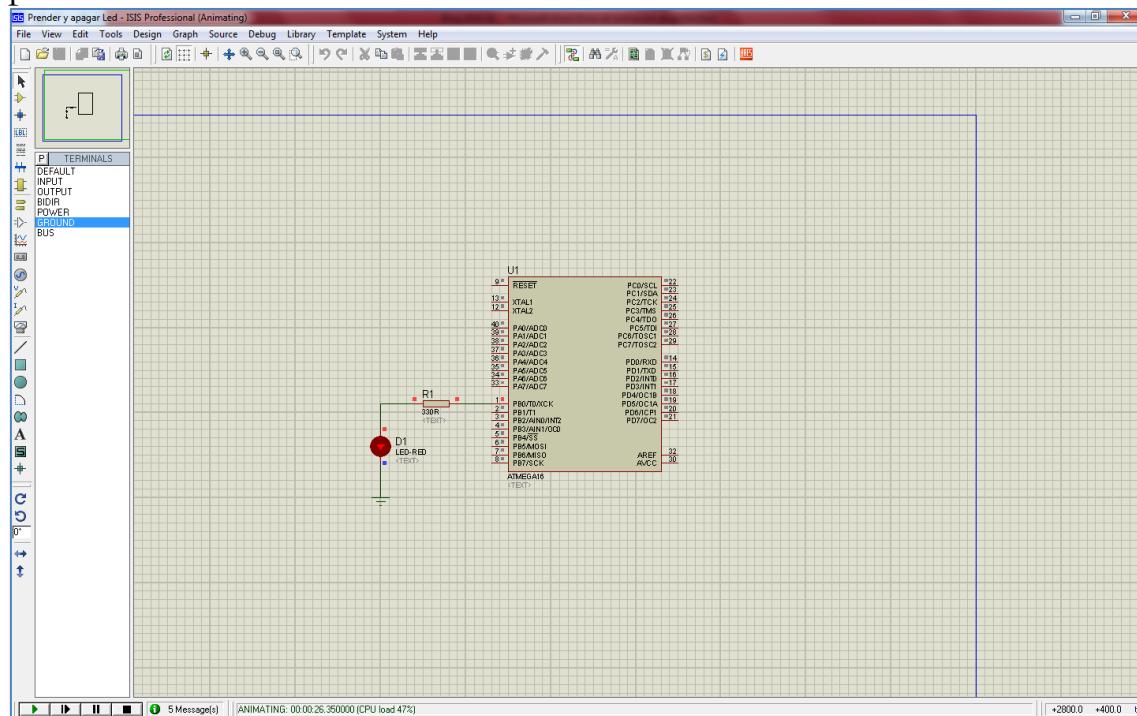


Figura 7

En la Figura8, se puede apreciar como la onda de activación del LED varía de “1” a “0”, haciendo que el Led se prenda y apague:

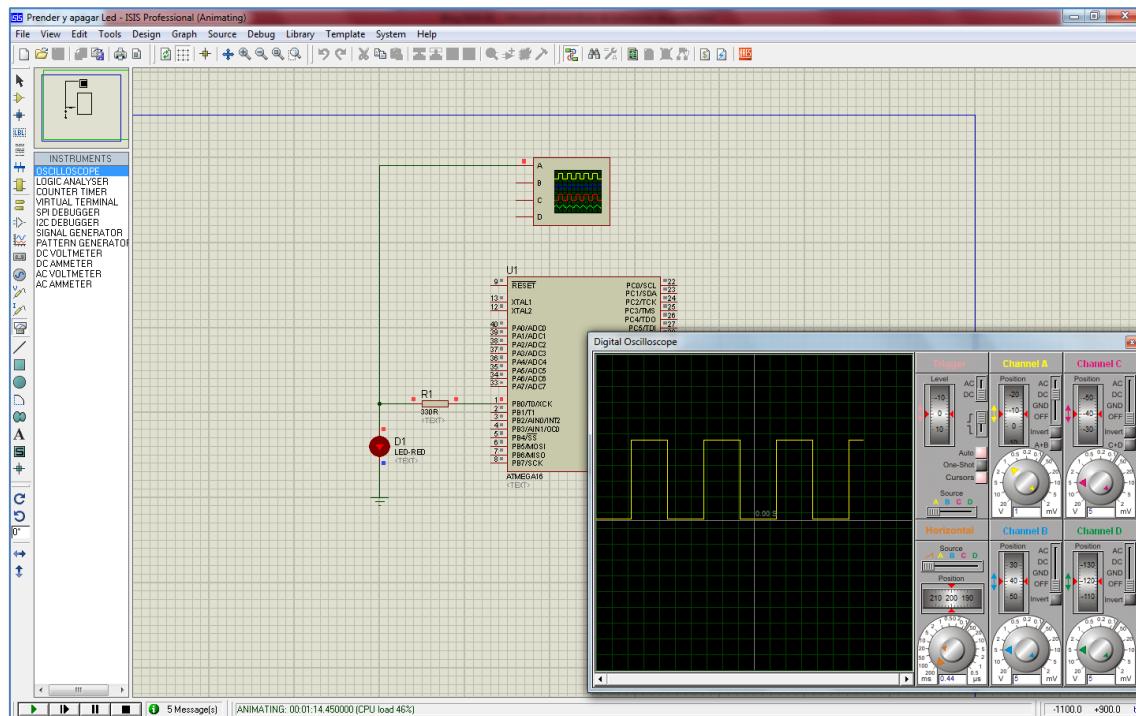


Figura8

En la Figura9, se puede muestra que el tiempo que permanece encendido y apagado es de 1segundo.

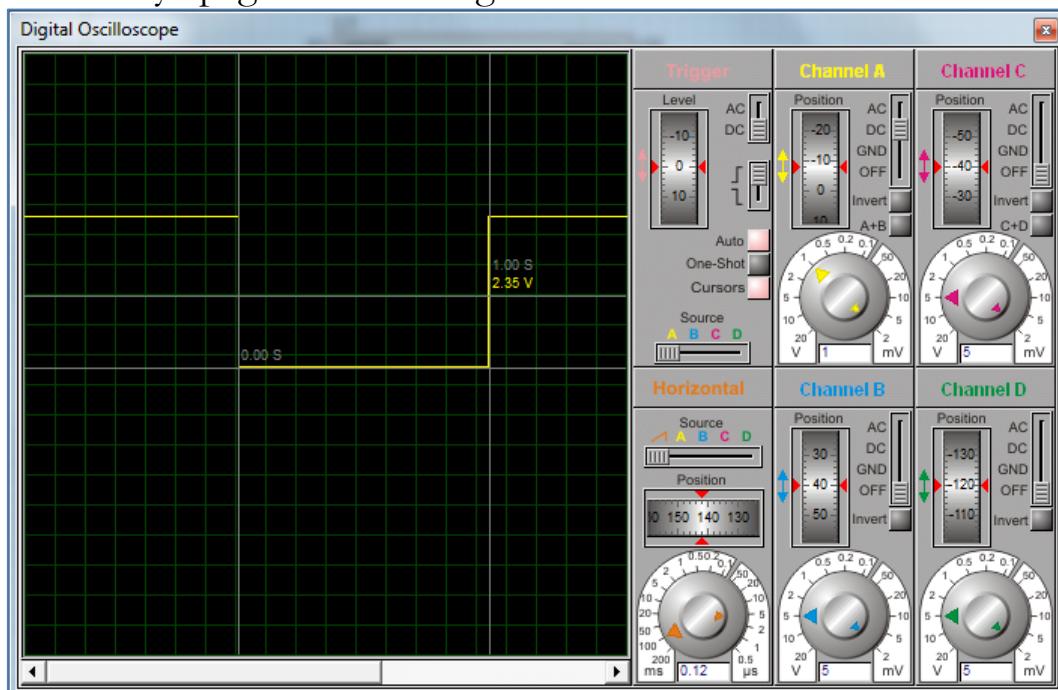


Figura9

En la Figura10, se da a conocer que cuando se puso a compilar el código del programa no se encontró ningún error.

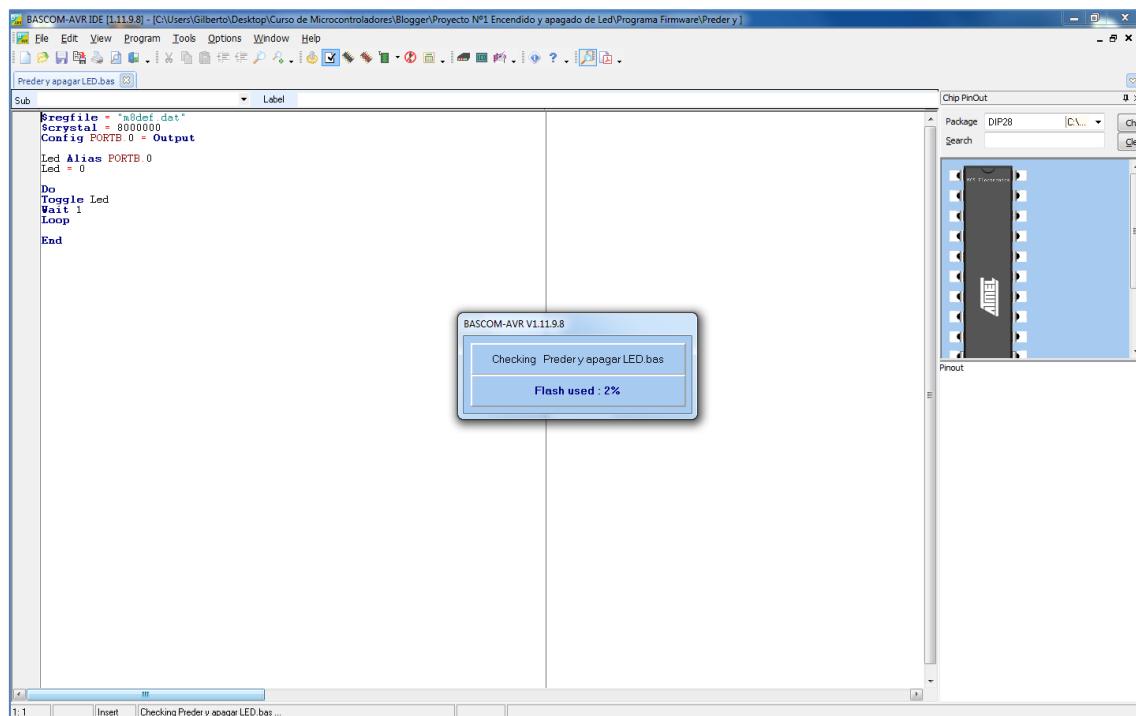


Figura10

“Si te gusto el artículo, compártelo, recuerda que hay otros que desean aprender y tú puedes ayudarlos también comentando y **compartiéndolo en tus redes**”.

¡Gracias!  
Atte. Gilberto Pascual