

¿CÓMO PROGRAMAR UN ATTINY84 CON ARDUINO COMO ISP EN LENGUAJE C?

La mayoría de las veces cuando se desea programar un uC de la familia ATtiny es necesario utilizar programadores/debuggers los cuales permitan realizar dicha acción, por lo regular se utiliza un dispositivo dedicado para ello, pero otras veces por falta de presupuesto o ya que se tienen otras herramientas a la mano es necesario acoplarse a la situación.

MATERIAL.

- **HARDWARE.**
 - Arduino (Elegoo) Mega 2560.
 - ATtiny84.
 - 1 LED.
 - Capacitor de 10uF.
 - Jumpers hembra macho.
 - Cables.
- **SOFTWARE.**
 - ArduinoIDE.
 - Microchip Studio.
 - AVRDUDESS.

PROCEDIMIENTO.

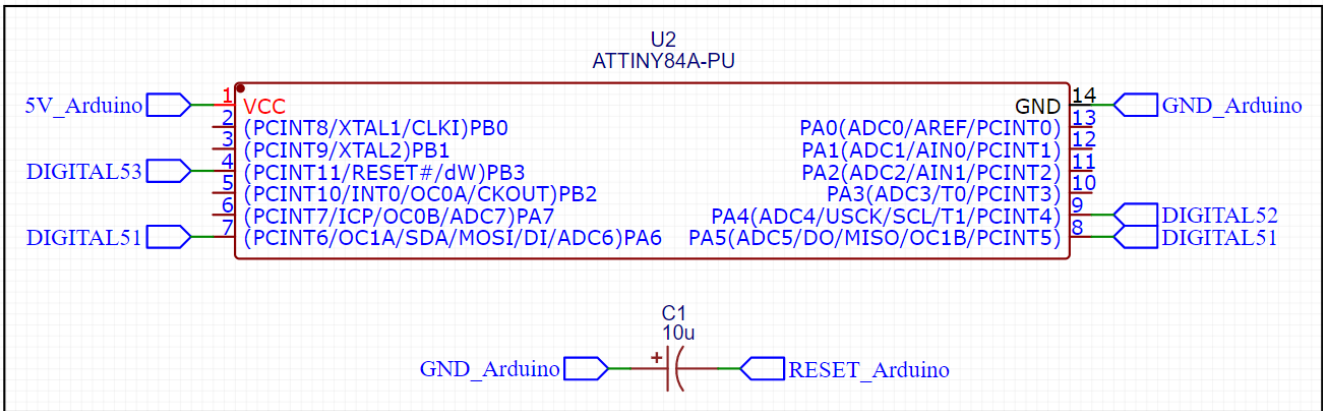
Como primer paso lo que se realizara es configurar el Arduino Mega 2560 como ISP (In-System Programmer), esto con la finalidad de que actúe como puente entre el programa a cargar y el uC. Para ello es necesario abrir el IDE de Arduino (ArduinoIDE) y en la parte superior en dirigirse a “[File > Examples > 11.ArduinoISP > ArduinoISP](#)” una vez realizado esto y conectado el Arduino en el puerto COM correspondiente es momento de ir a la misma barra superior y en “[Tools > Board > Arduino AVR Boards > Arduino Mega or Mega 2560](#)” se selecciona la placa correspondiente, ahora en “[Tools > Port > COM#](#)” se selecciona el puerto donde se encuentra la placa, de igual forma en “[Tools > Processor > ATmega2560 \(Mega 2560\)](#)” es posible seleccionar el procesador a utilizar y por último en “[Tools > Programmer > Arduino as ISP](#)” se selecciona como actuara la placa, con todo lo anterior configurado dentro del código de Arduino se realizarán unos pequeños cambios en la línea 73, 85, 86 y 87.

```
73 #define RESET 53
85 #define ARDUINOISP_PIN_MOSI 51
86 #define ARDUINOISP_PIN_MISO 50
87 #define ARDUINOISP_PIN_SCK 52
```

Realizados estos cambios se sube el archivo a la placa con la cual se esta trabajando.

Una vez configurada la placa de Arduino como ISP el siguiente paso es conectar el ATtiny84 con la placa de Arduino configurada.

CONEXIONES ARDUINO ATTINY84 Y CAPACITOR



Nota. La conexión del PIN8 del ATtiny84 va hacia el DIGITAL50 del Arduino.

Realizadas las conexiones correspondiente es hora de escribir el programa en C con el cual se trabajara, para ello es necesario abrir Microchip Studio y crear un nuevo proyecto en “**File > New > Project**”, nombrar el proyecto y elegir una dirección de guardado.

Una vez creado el proyecto se deberá escribir el siguiente código:

```
01 #define F_CPU 1000000UL
02 #include <avr/io.h>
03 #include <util/delay.h>
04 int main(void){
05     DDRA |= (1 << DDA0);
06     while(1){
07         PORTA |= (1 << PORTA0);
08         _delay_ms(1000);
09         PORTA &= ~(1 << PORTA0);
10         _delay_ms(1000);
11     }
12 }
```

Escrito el código es hora de construirlo, para ello es necesario ir a “**Build > Build Solution**”.

Creada la solución es hora de cargarla al uC, para ello es necesario utilizar el programa AVRDUDESS, y colocar los siguientes parámetros:

AVRDUDESS 2.18 (avrdude version 8.0)

Programmer (-c)
arduino_as_esp ... (AVR as programmer with Arduino-as-ISP FW)

Port (-P) Baud rate (-b) Bit clock (-B)
COM5

Flash
C:\Users\nikob\OneDrive\Escritorio\GRADUS\DPTO COMPUTO\Tesis\PRUEBAS\Attiny84\testLEDuChipStudio\VedBlinkC\VedBlinkC\Debug\VedBlinkC.hex

☒ Write ☐ Read ☐ Verify

MCU (-p)
ATtiny84

Flash: 8 KB 1E930C
EEPROM: 512 B

Presets
Default

Manager

Y listo, así de sencillo se programa un ATtiny84 con un Arduino como ISP.