

# PROYECTO A5: ATTINY85



por [Aurelio Gallardo](#)

Los ATtiny son microchips de la familia de Arduino. En realidad , su mínima expresión. El “8” del 85 indican que poseen 8Kb de memoria flash y el “5” indica que tienen 5 pines de propósito general más uno de RESET más VCC y GND. Pero lo más importante es su tamaño reducido: su encapsulado PDIP de 8 pines sólo ocupa 8x9 mm.

Por lo demás, su frecuencia de reloj alcanza los 20MHz, se puede comunicar por I2C y SPI y su rango de voltaje varía entre los 2.7V - 5.5V.

¿Qué podemos hacer con él? La respuesta es fácil: dispositivos muy pequeños. Eso sí... No tiene puertos USB así que ¿cómo lo programamos?

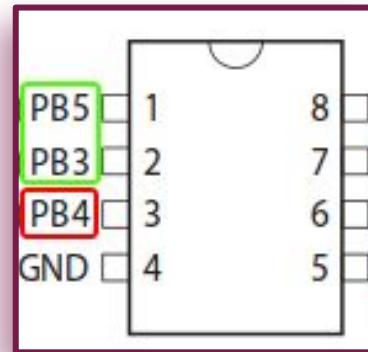


# Método de programación. Introducción

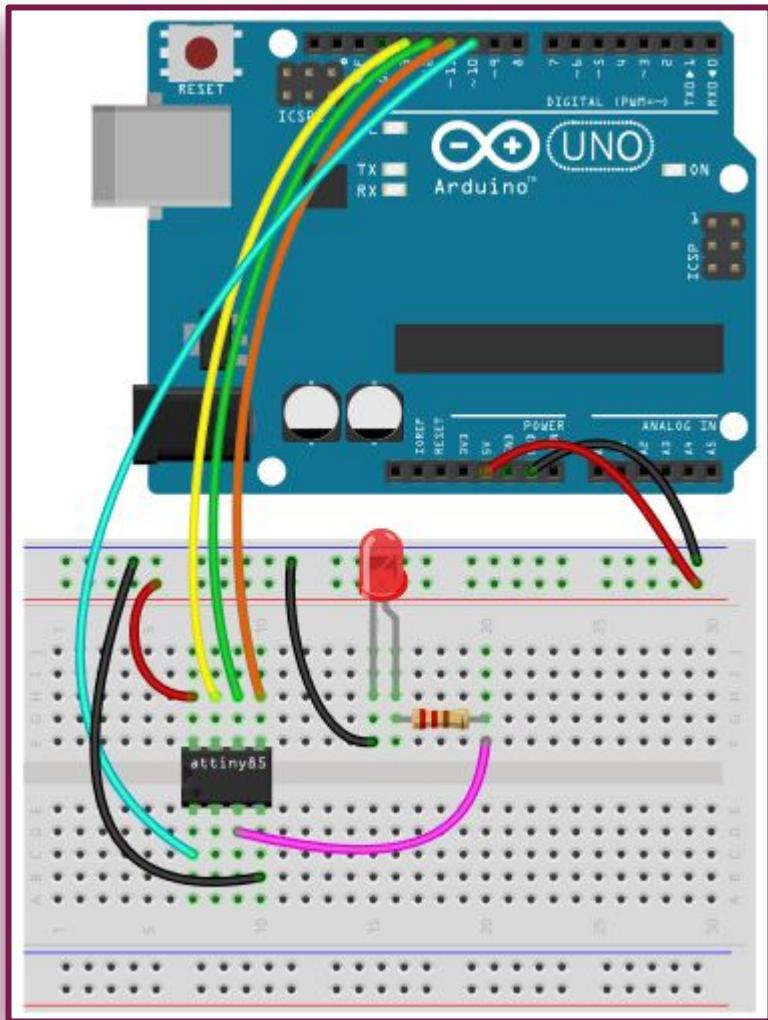
Para programar un Attiny85 necesitamos un programador. Podemos usar algunos que [encontraremos por internet](#), usar directamente una tarjeta [Digispark](#) (<https://www.luisllamas.es/programar-digispark-con-el-ide-de-arduino/>) o, lo que vamos a hacer nosotros, programarlo con un ARDUINO UNO como programador. Por ahora, nuestro objetivo es hacer un “Hola mundo” con nuestro ATtiny85.

Así que conecta nuestro ATtiny85 a nuestro Arduino UNO como en la figura de la página siguiente. Y escribe el programa del parpadeo. Nuestro ATtiny parpadeará en su pin PB4 (patillaje DIP-3 - En rojo).

Pero ¡¡Espera!! ¡¡Que no es tan sencillo!!  
Tienes que seguir los siguientes pasos.  
Dale un vistazo al esquema y al programa,  
pero léelo todo antes



# Método de programación. Blink “hola mundo”



```
const int LED = 4;

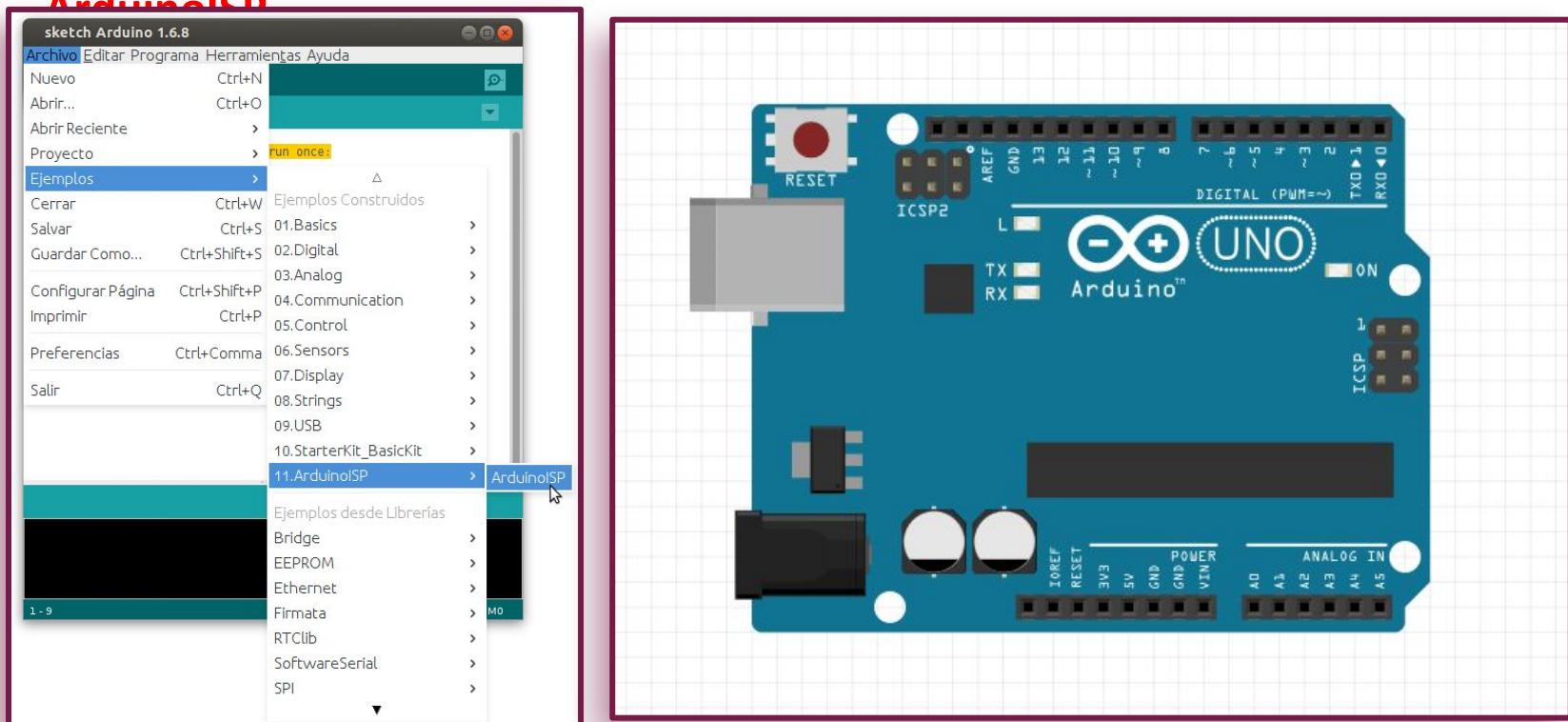
void setup() {
  pinMode(LED, OUTPUT);
}

void loop() {
  digitalWrite(LED, HIGH);
  delay(500);
  digitalWrite(LED, LOW);
  delay(500);
}
```

**Materiales:** ATtiny85, Arduino Uno, LED, 220 ohm. Ordenador con IDE de Arduino, placa protoboard, cable USB de conexión, cables jumper dupont 2.54 macho-macho. Para la práctica: LDR, Buzzer, resistencias de 10K

# Método de programación. Paso 1

Lo primero que tenemos que hacer para poder utilizar nuestro **Arduino UNO** como programador es grabarle el sketch para que se comporte como tal. Para ello, conectamos nuestro Arduino y abrimos Arduino IDE. El sketch que necesitamos se encuentra en los ejemplos de Arduino IDE, y lo podemos encontrar en **Archivo > Ejemplos > 11.ArduinoISP > ArduinoISP**.

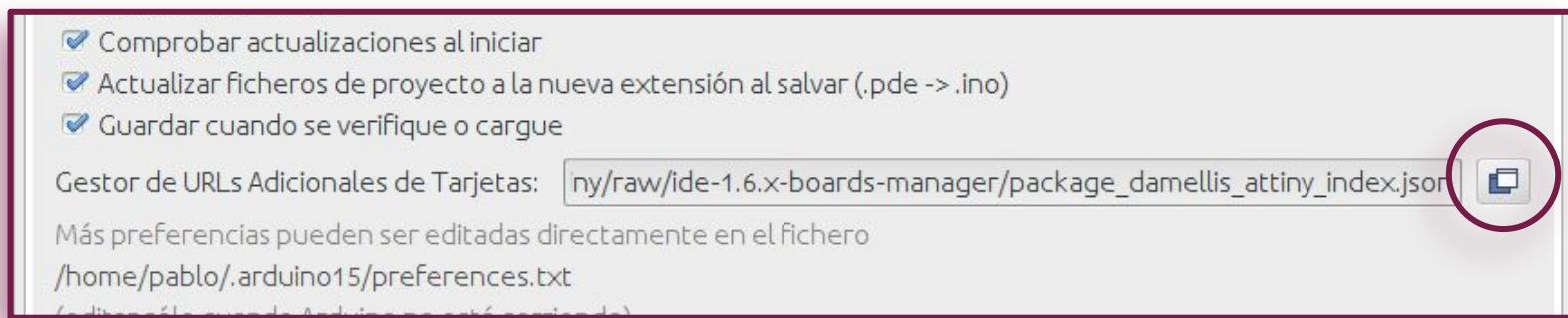


## Método de programación. Paso 2

Cuando programamos una placa Arduino o compatible, tenemos que seleccionarla en el menú **Herramientas > Placa** donde se abre una lista de placas que podemos programar. Si echas un vistazo en tu Arduino IDE, no encontrarás el ATtiny entre ellas.

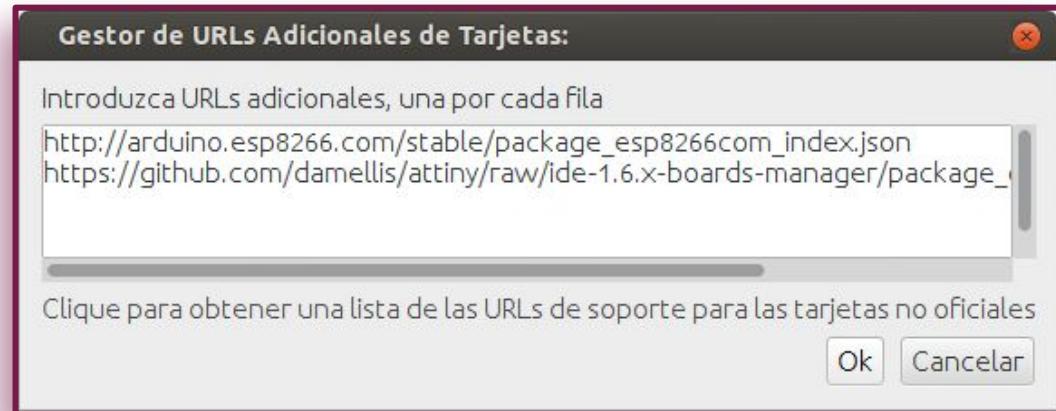
Para añadir soporte para ATtiny en Arduino IDE, tenemos que ir al menú **Archivo > Preferencias**, y en **Gestor de URLs Adicionales de Tarjetas** pegamos la siguiente URL:

[https://github.com/damellis/attiny/raw/ide-1.6.x-boards-manager/package\\_damellis\\_attiny\\_index.json](https://github.com/damellis/attiny/raw/ide-1.6.x-boards-manager/package_damellis_attiny_index.json)



## Método de programación. Paso 3

Si tenemos más URLs de otras tarjetas, podemos cliquear en el botón junto al cuadro de texto, se nos expandirá, y ahí podemos añadir una URL por cada línea.



Esta URL contiene información sobre los paquetes necesarios para instalar soporte para los ATtiny en Arduino IDE, pero no instala el software necesario. Para hacerlo, vamos al menú **Herramientas > Placa > Gestor de tarjetas...** y, en el diálogo que aparece y después de dejar que cargue toda la información, buscamos en la lista **attiny by David A. Mellis** y le damos a instalar.

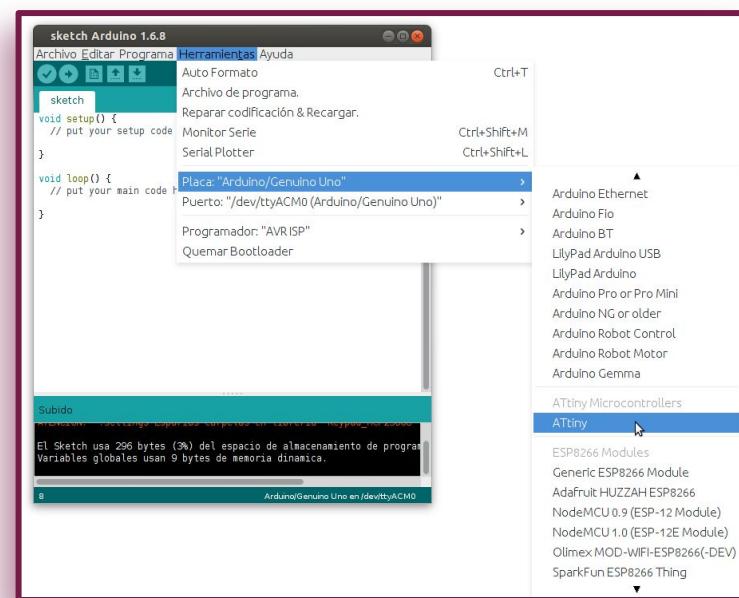


## Método de programación. Paso 4

Realizamos el conexionado con el ATtiny como en el esquema. Los pines 13 al 11 de Arduino UNO corresponden al puerto SPI del ATmega328P, que se conectan los pines SPI del ATtiny, mientras que el pin 10 se utiliza para activar el RESET de ATtiny.

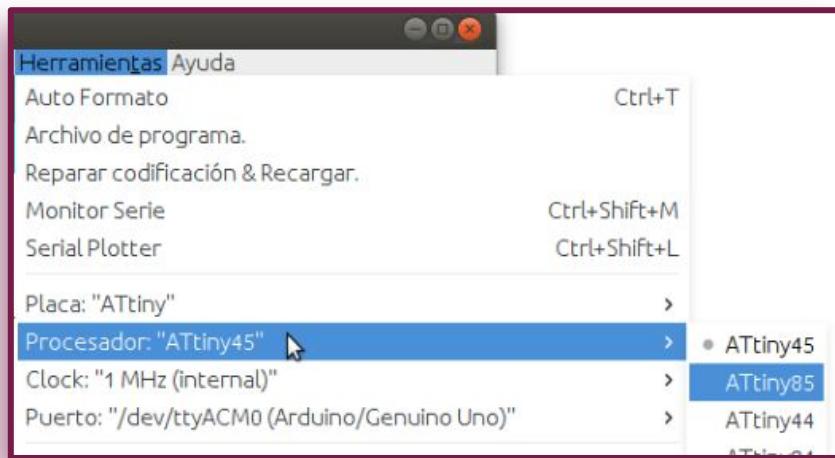
**NOTA:** Algunas placas requieren un condensador de 10uF entre el pin RESET y GND de tu placa Arduino. Si tienes problemas, prueba a añadirlo a tu circuito.

**Configurando:** Lo primero que vamos a hacer es, en Arduino IDE, ir a Herramientas > Placa y buscar ATtiny.



## Método de programación. Paso 5

Seleccionamos esta configuración para el ATtiny85 (Se puede poner de reloj 8MHz pero que sea INTERNO; si no lo hacemos, y lo ponemos externo, no funcionará como no le pongamos un oscilador externo).



Programador: "Arduino as ISP"  
Quemar Bootloader



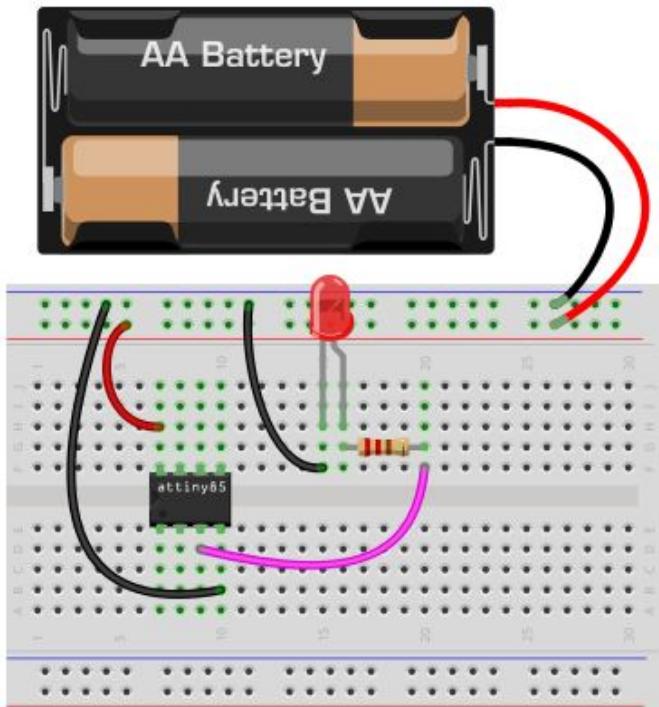
Selecciona ahora en programador “ARDUINO as ISP” (no otro). **Quema el bootloader.**

Y ya, ve al programa **blink** y **sube el sketch**. Nuestro Arduino UNO hará “de puente” y entregará el programa al ATtiny 85. Ya puedes desconectarlo todo y volver a conectarlo simplemente con el led en el pin 4 y la resistencia. Aliméntalo... Y verás que funciona.

# Método de programación. Alimentación

Todas estas posibilidades son factibles con ATtiny para alimentarlo. ¡ Lo más importante es no sobrepasar los 5.5V !

- 2 pilas AA en serie de 1,5V cada una (3V en total).
- Una pila de botón de 3V.
- Una pila de petaca de 4,5V (que es lo mismo que 3 pilas AA en serie).
- Una batería de litio de 3,7V.



Información obtenida de la web  
[fabricadigital.org](http://fabricadigital.org/)

<https://fabricadigital.org/tutoriales/utiliza-tu-placa-arduino-como-programador-para-attiny/>

<https://fabricadigital.org/tutoriales/blink-con-attiny-y-arduino-ide/>



## Ejercicio A: la alarma de cajón

Sospecho que en casa un ladronzuelo de cuatro años me abre los cajones. Creo que le voy a dar un buen susto con una alarma que suene cuando abra un cajón. El método es sencillo: una ldr, ajustada convenientemente, activa un buzzer cuando incide la luz sobre ella.

¿Te atreves a hacerla con un ATtiny85?

