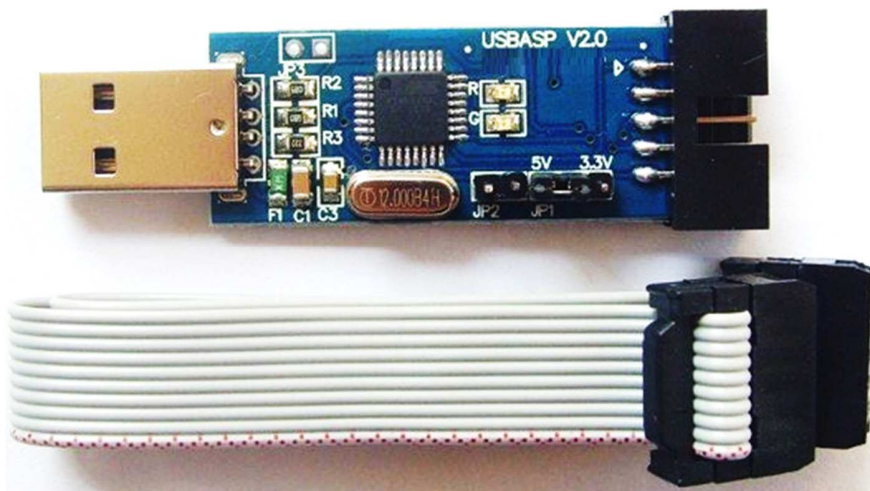
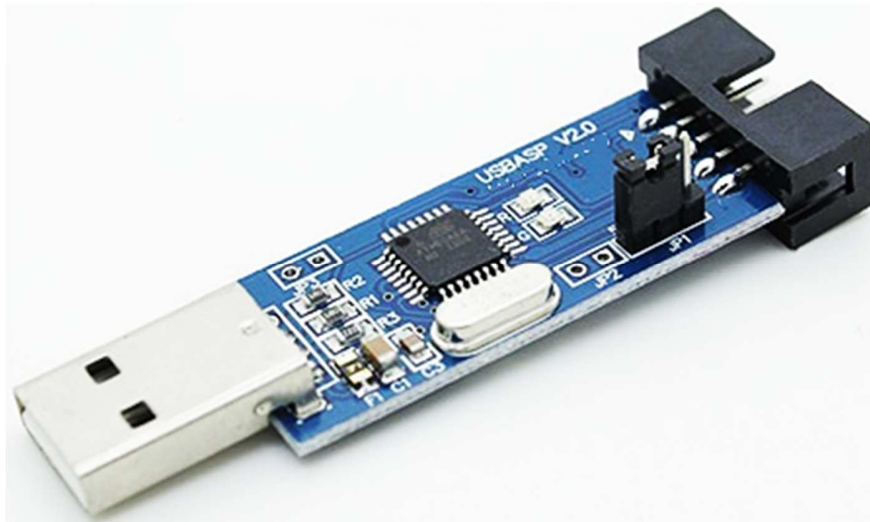


V 1.0

# MANUAL DE USUARIO PROGRAMADOR USBASP AVR



# TABLA DE CONTENIDO

<b>INTRODUCCIÓN .....</b>	<b>3</b>
Microcontroladores soportados .....	3
<b>DESCRIPCIÓN .....</b>	<b>4</b>
1. Entrada de voltaje USB .....	4
2. Regulador a 3.3V.....	5
3. Pines de control.....	5
4. Pines de salida .....	6
5. Indicador de encendido y programación.....	7
Resumen.....	7
<b>FUNCIONAMIENTO .....</b>	<b>7</b>
Paso uno: Descargar lo necesario. ....	8
Paso dos: Abrir el Software de programación.....	10
Paso tres: conexión del micro a los pines ISP - SPI. ....	12
Paso cuatro: Programación.....	13

## INTRODUCCIÓN

USBASP es un programador USB para los controladores AVR Atmel. Simplemente consiste en un ATmega8 y algunos componentes pasivos.

El programador utiliza un controlador USB-firmware solamente, no se necesita ningún controlador USB especial.

Algunas de las características clave incluyen:

- Funciona bajo múltiples plataformas. Linux, Mac OS X y Windows fueron probadas.
- La programación a una velocidad de hasta 5kB/seg.
- Opción SCK lento para apoyar los objetivos con velocidad de reloj bajo (<1.5 MHz).

## Microcontroladores soportados

La Tabla 1 muestra los microcontroladores que son compatibles con el Programador USBASP AVR.

Supported Microcontrollers				
Mega Series				
ATmega8	ATmega8A	ATmega48	ATmega48A	ATmega48P
ATmega48PA	ATmega88	ATmega88A	ATmega88P	ATmega88PA
ATmega168	ATmega168A	ATmega168P	ATmega168PA	ATmega328
ATmega328P	ATmega103	ATmega128	ATmega128P	ATmega1280
ATmega1281	ATmega16	ATmega16A	ATmega161	ATmega162
ATmega163	ATmega164	ATmega164A	ATmega164P	ATmega164PA
ATmega169	ATmega169A	ATmega169P	ATmega169PA	ATmega2560
ATmega2561	ATmega32	ATmega32A	ATmega324	ATmega324A
ATmega324P	ATmega324PA	ATmega329	ATmega329A	ATmega329P
ATmega329PA	ATmega3290	ATmega3290A	ATmega3290P	ATmega64
ATmega64A	ATmega640	ATmega644	ATmega644A	ATmega644P
ATmega644PA	ATmega649	ATmega649A	ATmega649P	ATmega6490
ATmega6490A	ATmega6490P	ATmega8515	ATmega8535	
Tiny Series				
ATtiny12	ATtiny13	ATtiny13A	ATtiny15	ATtiny25
ATtiny26	ATtiny45	ATtiny85	ATtiny2313	ATtiny2313A
Classic Series				
AT90S1200	AT90S2313	AT90S2333	AT90S2343	AT90S4414
AT90S4433	AT90S4434	AT90S8515		
AT90S8535				
Can Series				
AT90CAN128				
PWN Series				
AT90PWM2	AT90PWM3			

Tabla 1. Referencias soportadas por el programador

**Advertencia:** Algunos de los componentes descritos en este documento son muy sensibles a las descargas estáticas eléctricas. El lector debe tomar precauciones para asegurarse de que los componentes están protegidos contra estas descargas.

Mientras que las tensiones suelen verse en los circuitos de microcontroladores son bajos, el lector debe tener en cuenta el riesgo de trabajar con circuitos eléctricos y tomar las precauciones necesarias.

## DESCRIPCIÓN

Se trata de una tarjeta programadora de referencias de AVR por medio de SPI<sup>1</sup> y conexión USB al PC.:

### 1. Entrada de voltaje USB

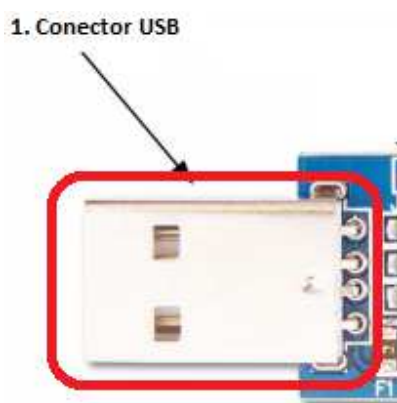


Figura 1: Pines entrada de voltaje.

Este conector USB cuenta con las señales de **5V** de entrada **GND**, **D+** y **D-**. De los cuales se utilizan los **5V** y **GND**, para la alimentación de la interfaz SPI para la programación.

**Advertencia:** Se debe tener mucha precaución en no invertir la polaridad del voltaje de entrada, puesto que puede ocasionar daños en la tarjeta.

<sup>1</sup> **SPI** es un estándar de comunicaciones, usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos. El bus de interfaz de periféricos serie o bus SPI es un estándar para controlar casi cualquier dispositivo electrónico digital que acepte un flujo de bits serie regulado por un reloj.

## 2. Regulador a 3.3V

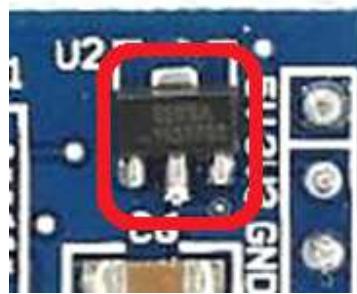


Figura 2: Regulador de 3.3V.

Este regulador permite tomar el voltaje de entrada (entre 5V), y arrojar 3.3V a la parte del jumper de selección **JP1**.

## 3. Pines de control

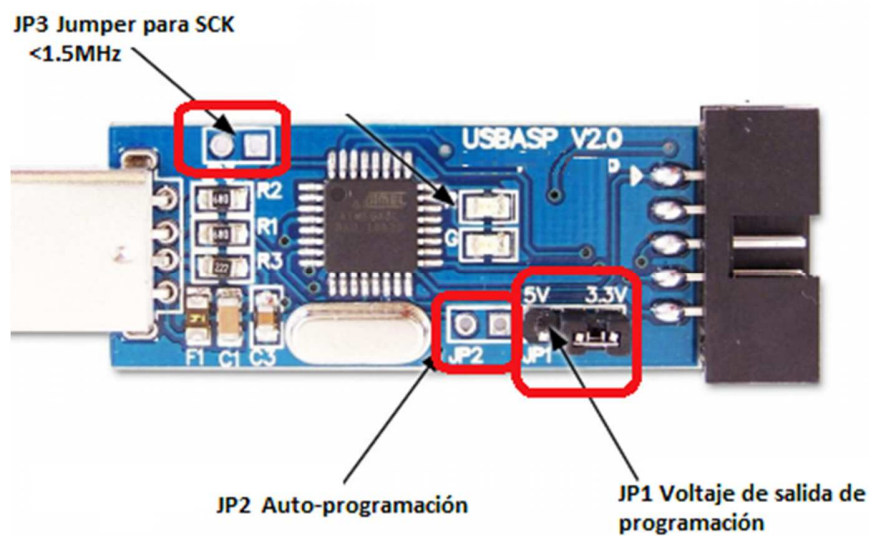


Figura 3: Jumpers de control

La tarjeta cuenta con tres jumpers:

- **JP1 – Selección de voltaje de salida**

Este puente controla la tensión en el conector ISP VCC. Se puede ajustar a +3,3 V, +5 V o desactivar este puente si el dispositivo de destino tiene su propia fuente de alimentación.

- **JP2 – Auto-programación**

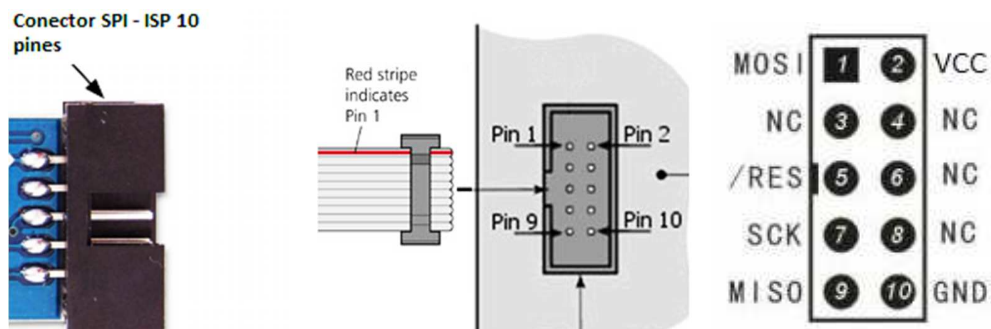
Este puente se utiliza para actualizar el firmware del programador USBASP. Con el fin de actualizar el firmware se necesitan 2 programadores. Uno a programar y que el otro haga la programación.

- **JP3 – “lento” SCK**

Cuando se selecciona este puente, el modo de reloj lento se habilita. Si el reloj de destino es inferior a 1,5 MHz, es necesario establecer este puente. Entonces SCK se escala hacia abajo desde 375 kHz hasta aproximadamente 8 kHz.

**ALGO PARA RESALTAR ES QUE A DIFERENCIA DE MUCHOS PROGRAMADORES, ESTE NO NECESITA VOLTAJE DE ALIMENTACIÓN PARA LA PROGRAMACIÓN, A MENOS QUE EL DISPOSITIVO YA LO TENGA INCLUIDO.**

#### 4. Pines de salida



**Figura 4: Pines de salida.**

La tarjeta cuenta con un conector de 10 pines, donde podemos encontrar los pines de programación por SPI – ISP.

## 5. Indicador de encendido y programación

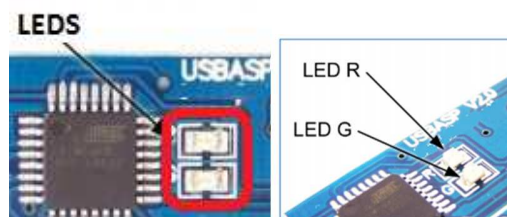


Figura 5: Led de encendido.

El programador USBASP tiene 2 LEDs cerca de la conexión ISP. Estos tienen las siguientes funciones:

- LED R - Programador comunicación con dispositivo de destino
- LED G – Alimentación.

Nota: Mientras que usted puede asumir que el LED R es rojo y LED G es verde, esto varía dependiendo del lote. Por ejemplo, dispone de 2 LEDs rojos.

## Resumen

- Le permite leer o escribir en la EEPROM del microcontrolador, firmware, fuse bits y bloquear bits soportado para Windows, Mac OS X y Linux (funciona en Windows 8.1)
- 5 KB / s Velocidad máxima de escritura
- Opción SCK Software controlada para apoyar los objetivos con velocidad de reloj de baja (<1.5 MHz)
- Interfaz ISP 10 pines (conforme a la norma ISP de 10 pines)

## FUNCIONAMIENTO

Ahora trataremos de realizar un ejemplo básico de funcionamiento, el cual describiremos por paso, el objetivo es mostrar el funcionamiento de la tarjeta.



## Paso uno: Descargar lo necesario.

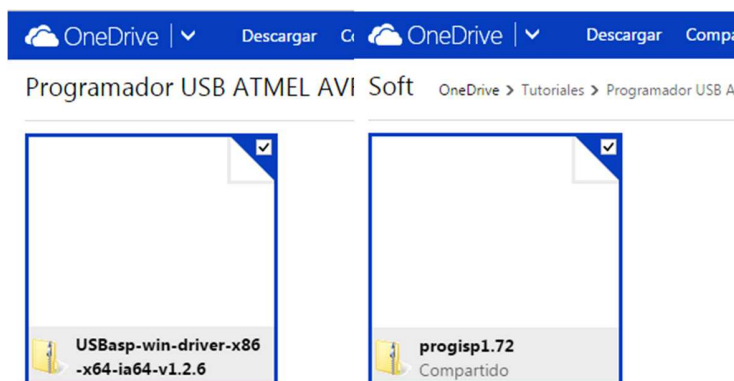


Figura 6: Archivos necesarios para el uso del Programador.

El primer link son los drivers, que se requieren para que se reconozca el programador: <http://1drv.ms/1zlj3O>

El segundo link es el software para la programación: <http://1drv.ms/1zljtUT>

Luego de descargar los archivos, os descomprimos y los dejamos en una carpeta o ruta sencilla, como el escritorio o mis documentos etc..

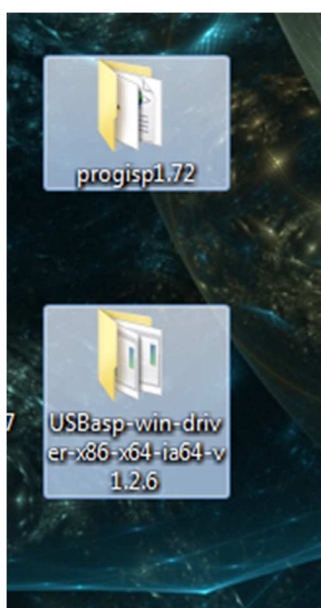
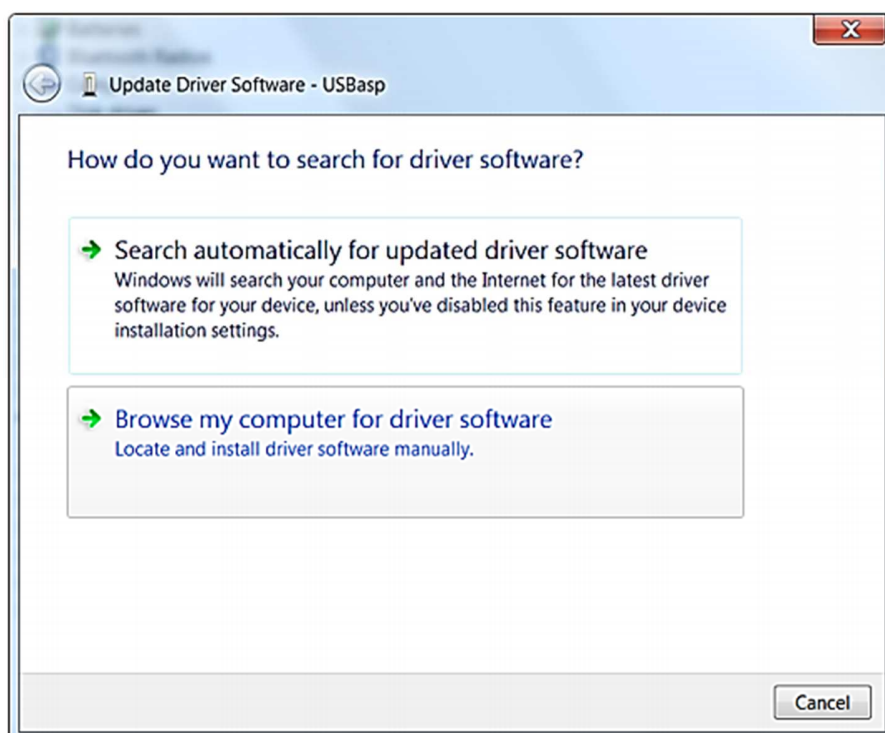
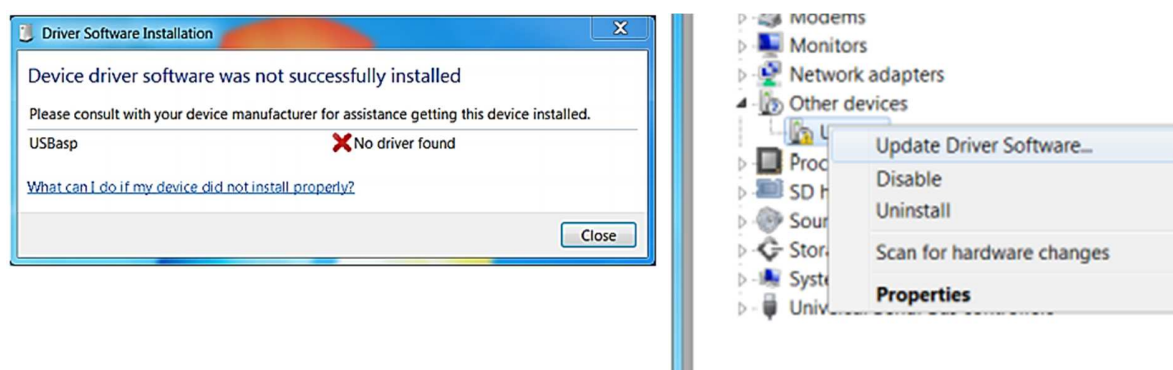
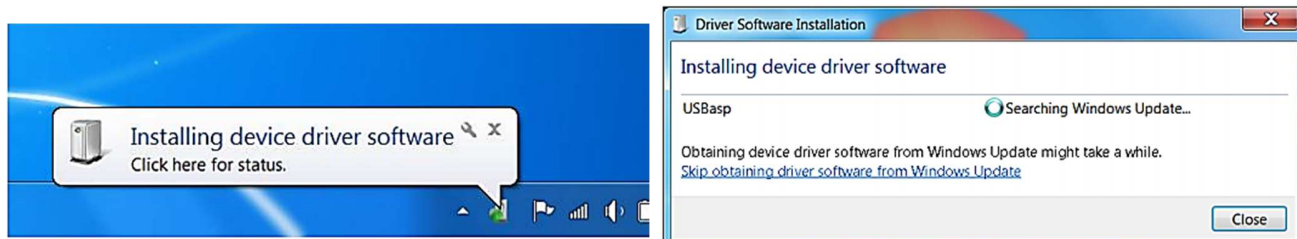
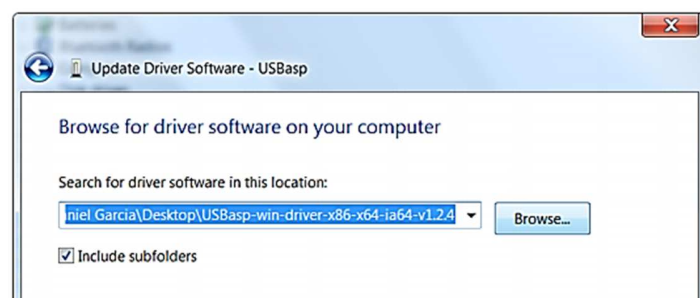


Figura 7: Ubicación de la carpeta.



Ahora vamos a conectar nuestro programador, y seguir lo paso de la instalación, el driver se encuentra en nuestra carpeta.





## Paso dos: Abrir el Software de programación

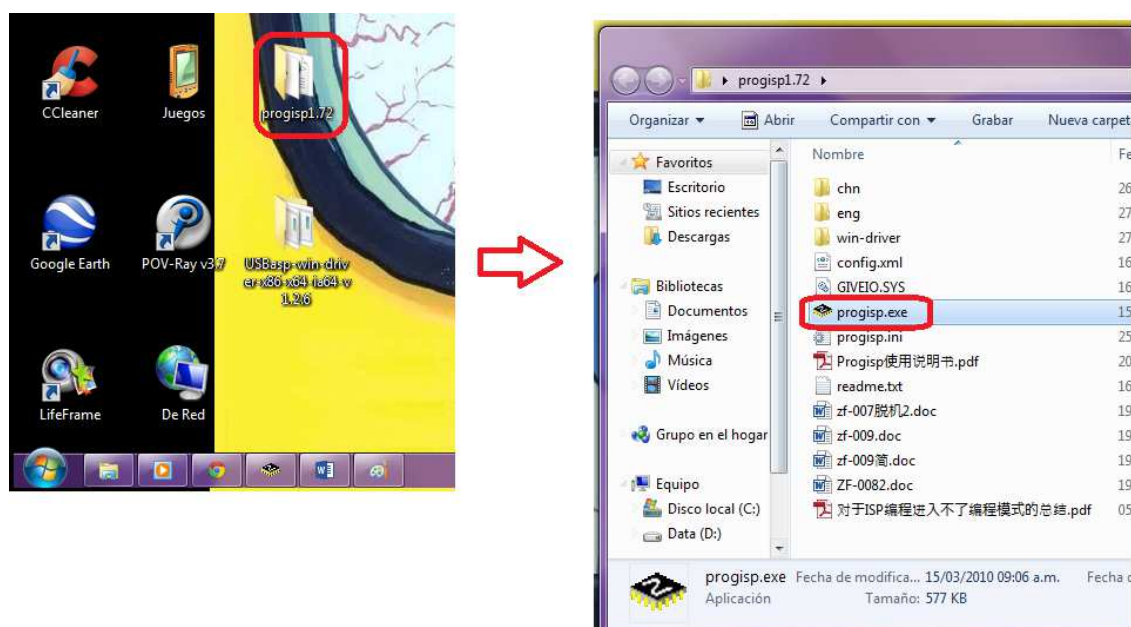


Figura 8: Software “progisp”

Vamos a abrir el software “progisp.exe”, luego veremos que se muestra en la figura 10.

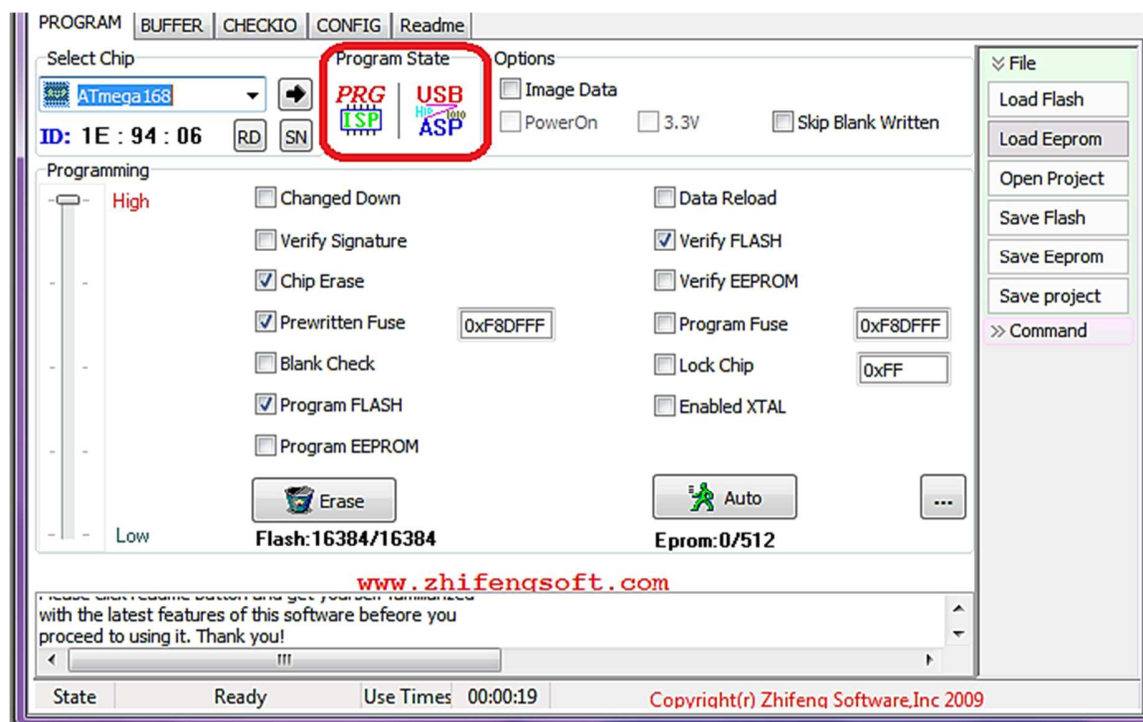


Figura 9: Interfaz de programación.

Allí se podemos ver la interfaz utilizada para la programación, en donde se detectara automáticamente el tipo de programador “USBASP”, y la forma de programación “PRG ISP”.

Esta interfaz cuenta con varias opciones, de las cuales solo utilizaremos, las que están seleccionadas.

### Paso tres: conexión del micro a los pines ISP - SPI.

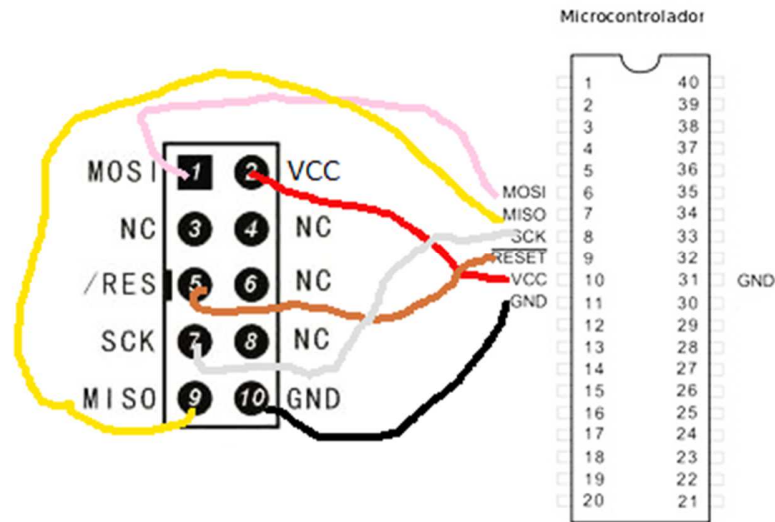


Figura 10: Conexión del programador al micro.

Para nuestro caso utilizaremos un ATMEGA 168, entonces debemos realizar la conexión como se muestra en la figura 10, aunque el software también nos sugiere como se debe conectar el micro según la referencia , puesto no todos se conectan de la misma manera, en la figura 12 podemos ver como se debe conectar según el software.

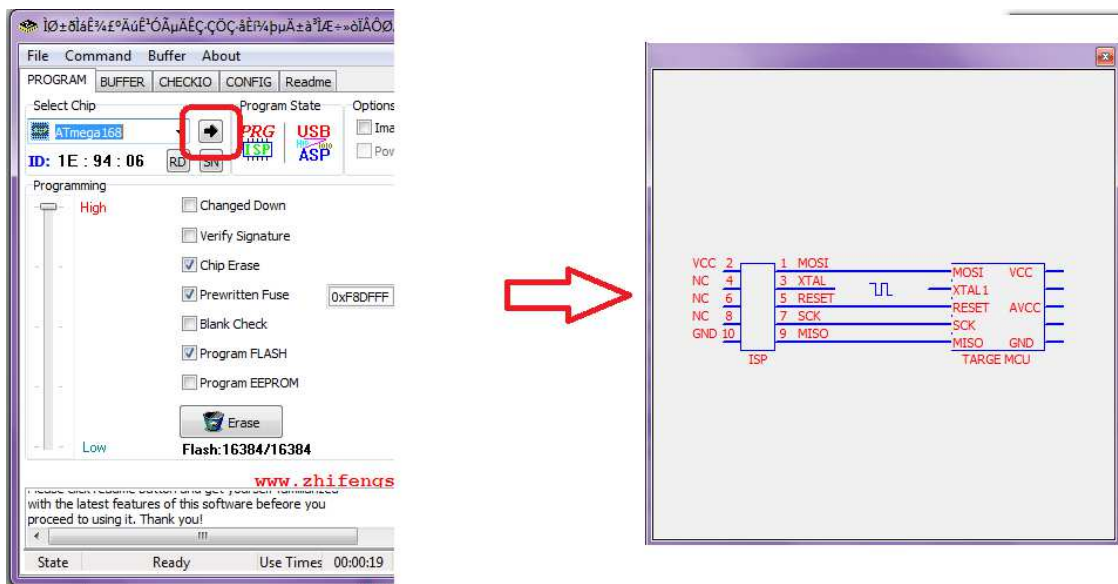


Figura 11: Conexión sugerida del programador al micro.

## Paso cuatro: Programación.

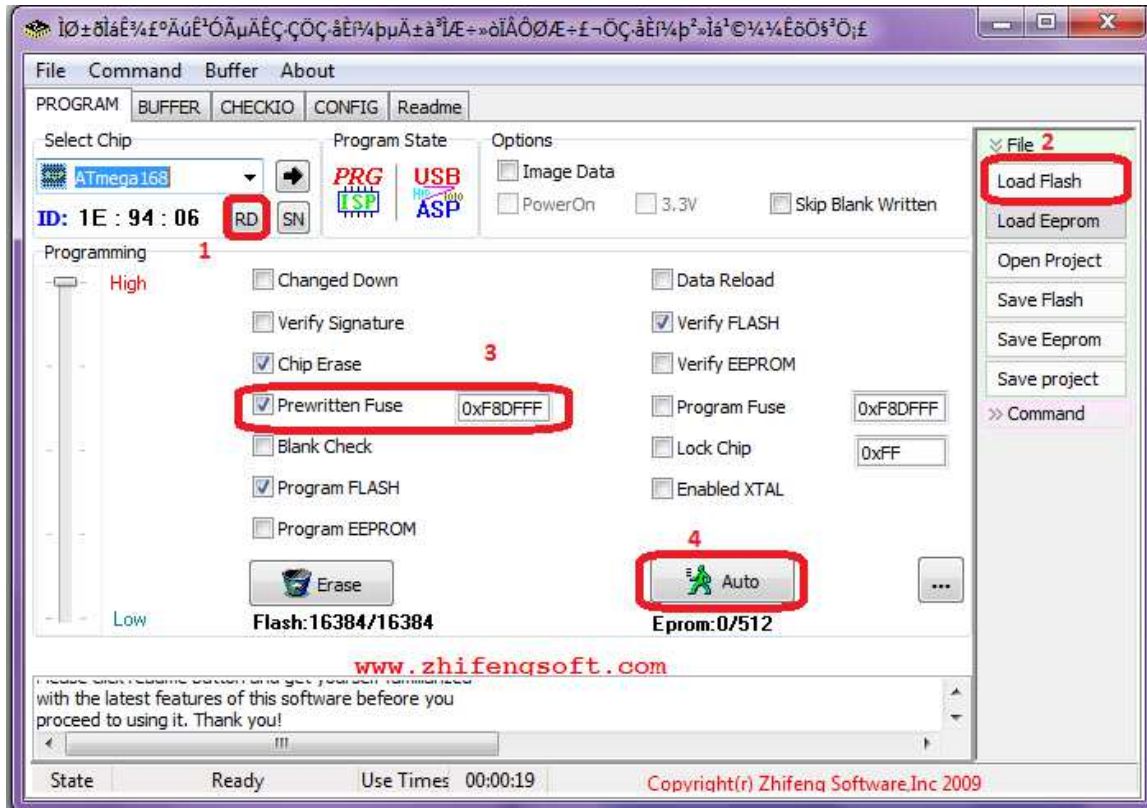


Figura 12: Pasos para programar.

Básicamente para la parte de programación debemos tener en cuenta los siguientes pasos:

Seleccionar con el jumper **JP1** el voltaje del micro a programar.

Realizar las conexiones de los pines de programación de micro a los pines de programación el **USBASP**.

Seleccionar en el software el micro a programar.

Hacer clic en el botón (1) **“RD”**, para que el software identifique que se trata del micro correcto el conectado con el seleccionado.

En la opción (2) **“Load Flash”**, se debe seleccionar el archivo **.HEX**, que se va a programar en el micro.

Se deben verificar los (3) **"FUSE"**, un aspecto muy importante.

Por ultimo antes de seleccionar el botón (4) **"Auto"**, verificamos que estén seleccionadas las casillas que se presentan en la figura 12, en la interfaz.