



*Aguijón*

---

*Notas de aplicación*

## **Nota de aplicación 18:**

### Read ADC and Send to RS232

#### **Descripción:**

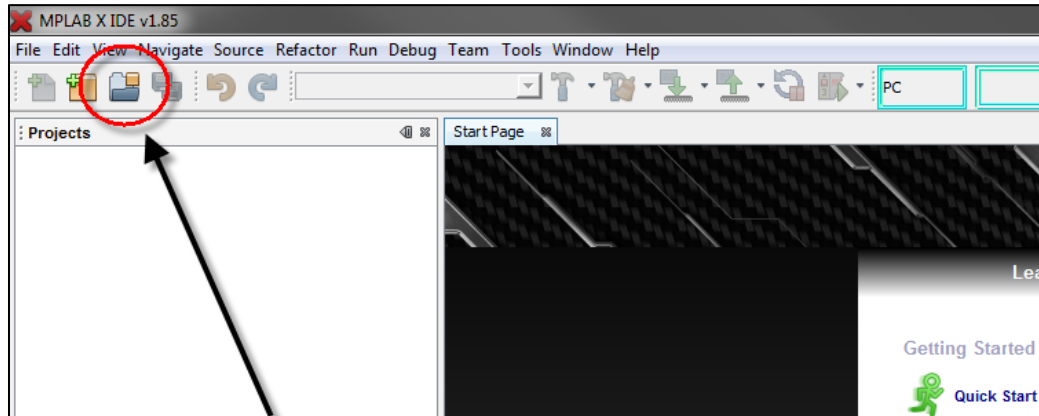
Leer el valor del ADC y mandarlo dicho valor por puerto serial (RS232).

#### **Herramientas:**

1. Aguijón 3.0, Aguijón 4.0 ó Aguijón 4.1
2. MPLAB X®
3. Aguijón HID bootloader
4. Cable USB 'A' to 'A'
5. Librerías HammerHead.
6. Cable convertidor USB - Serial

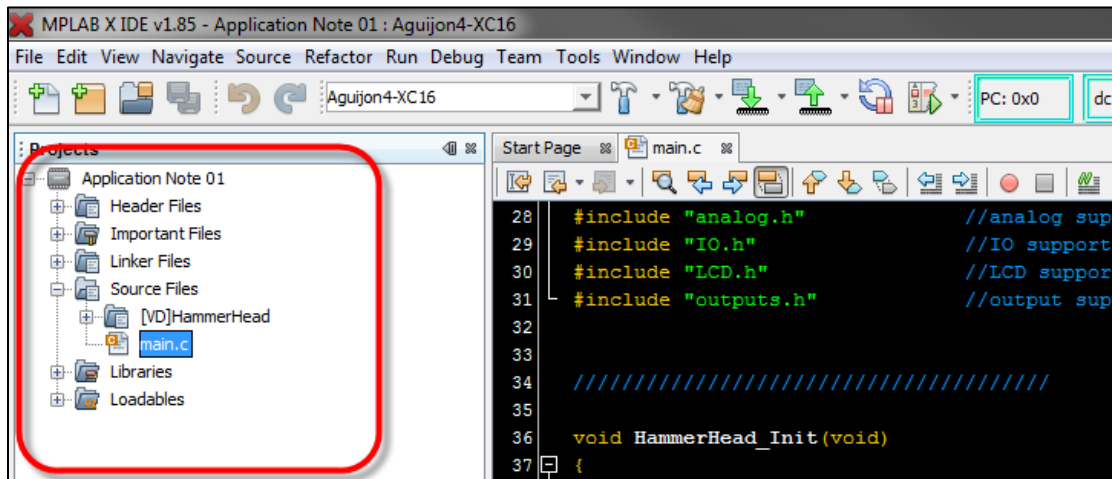
## Pasos:

1. Abrir MPLAB X® y cargar el archivo del proyecto: **Application Note 18.X**



Haz 'clic' aquí y  
abre el proyecto

1. Abrir el archivo **main.c**



2. Ir a la línea #81

Utilizaremos la siguiente función:

```
74     LCD_PutStr(1,0,"Vinagron Digital",TRUE);
75     LCD_PutStr(2,0,"Application Note 18",FALSE);
76
77     /* Process Phase */
78
79     for(;;){
80
81         adcReading = ADC_Read(); // Read ADC Input
82
83         sprintf(lcdMSG,"ADC Value: %i ",adcReading);
84         sprintf(rs232MSG,"ADC Value: %i \r",adcReading);
85         LCD_PutStr(1,0,lcdMSG,ON); // Puts ADC value
86         RS232_PutStr((unsigned int*)rs232MSG); // Puts ADC value
87
88         delayms(150);
```

**ADC\_Read ();**

- Esta función lee el puerto ADC;  
Regresa un valor equivalente al Valor análogo introducido (entero del 0 al 1024)

3. Ir a la línea #85

Utilizaremos la siguiente función:

```
78
79     for(;;){
80
81         adcReading = ADC_Read();           // Read ADC Input
82
83         sprintf(lcdMSG,"ADC Value: %i ",adcReading);
84         sprintf(rs232MSG,"ADC Value: %i  \r",adcReading);
85         LCD_PutStr(1,0,lcdMSG,ON);          // Puts ADC value
86         RS232_PutStr((unsigned int*)rs232MSG); // Puts ADC value
87
88         delayms(150);
89
90     }
91
92     return 0;
```

**LCD\_PutStr (int y, int x, char \*msg, BOOL clear);**

- Función que muestra una cadena de caracteres en la pantalla LCD; donde:  
**Int y** = Coordenada en y (Valor entero del 1 al 2.)  
**Int X** = Coordenada en x(Valor entero del 0 al 20.)  
**Char \*msg** = Cadena de caracteres (De 0 a 20 caracteres)  
**BOOL clear** = Determina si se borra la pantalla antes de escribir  
(TRUE =Borrar, FALSE =No borrar).

4. Ir a la línea #90

Utilizaremos la siguiente función:

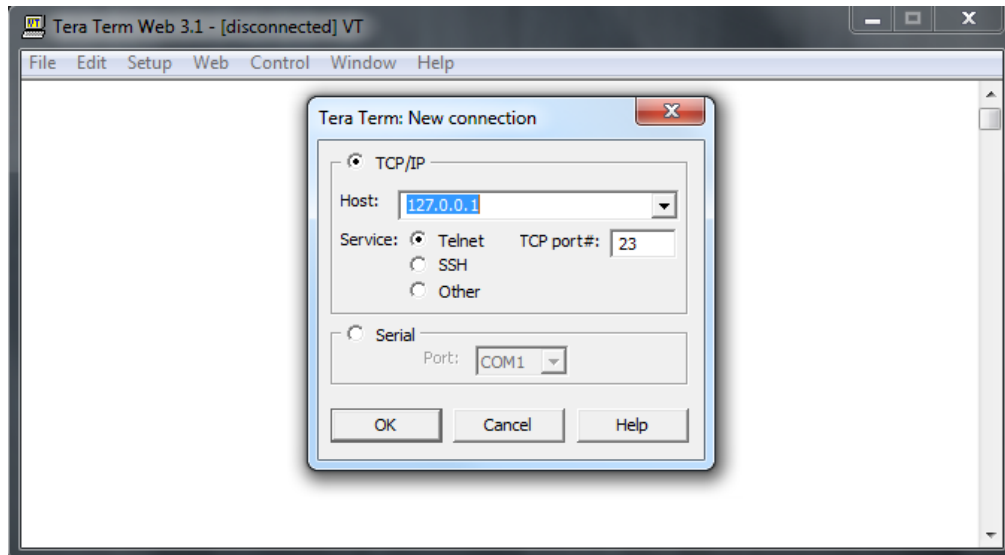
```
79     for(;;){
80
81         adcReading = ADC_Read();           // Read ADC Input
82
83         sprintf(lcdMSG,"ADC Value: %i ",adcReading);
84         sprintf(rs232MSG,"ADC Value: %i  \r",adcReading);
85         LCD_PutStr(1,0,lcdMSG,ON);         // Puts ADC value
86         RS232_PutStr((unsigned int*)rs232MSG); // Puts ADC value
87
88         delayms(150);
89
90     }
91
92     return 0;
93 }
```

**RS232\_PutStr(unsigned int \*string);**

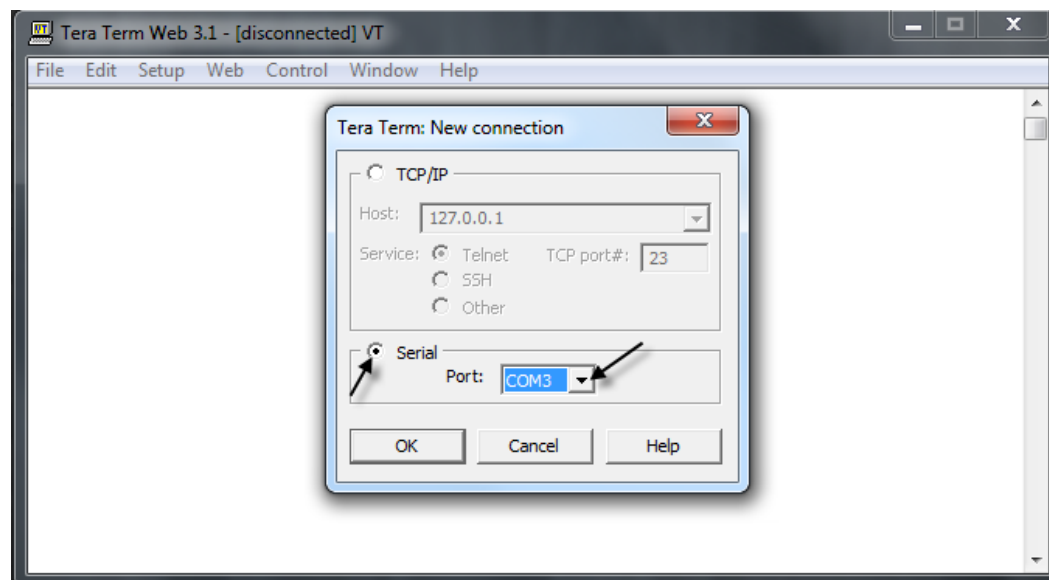
- Función que nos permite mandar una cadena de caracteres al puerto serie por medio del protocolo RS232; Donde:

**Unsigned int \*String=** Cadena de caracteres a enviar por el puerto serial.

5. Abriremos cualquier emulador de terminal que tengamos instalado, en este caso utilizaremos TERA TERM. Si no cuentas con un emulador de terminal puedes bajar TERA TERM de la siguiente dirección: <http://www.teraterm.org/>



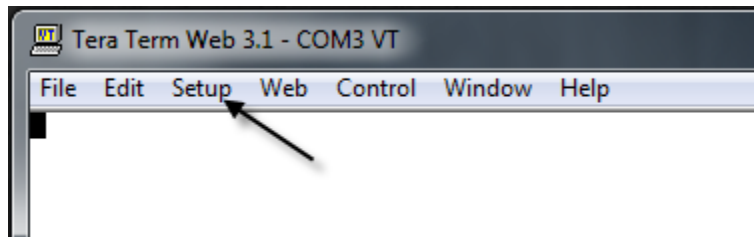
6. Conectaremos nuestro cable USB – Serial y seleccionamos el COM que nuestro cable este utilizando. En el ejemplo utilizaremos COM 3.



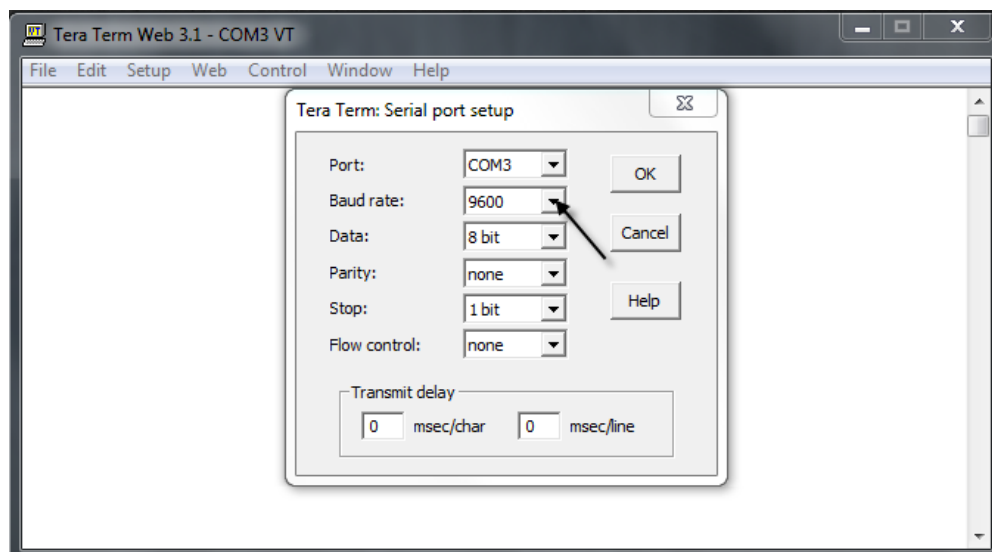
7. Después ajustamos los bits por segundo de nuestro emulador de terminal, tienen que ser los mismos valores de los que tenemos en nuestro código, para comprobar la velocidad de nuestro código iremos a la línea #46 del archivo **main.c**

```
43     ADC_Init();  
44  
45     /*UART*/  
46     RS232_Init(BAUD9600); //set 9600bps as UART speed  
47  
48     /*LCD*/  
49     LCD_Init(LCD_MODE_1);
```

8. Para luego ajustar los bits por segundo de nuestro emulador de terminal.

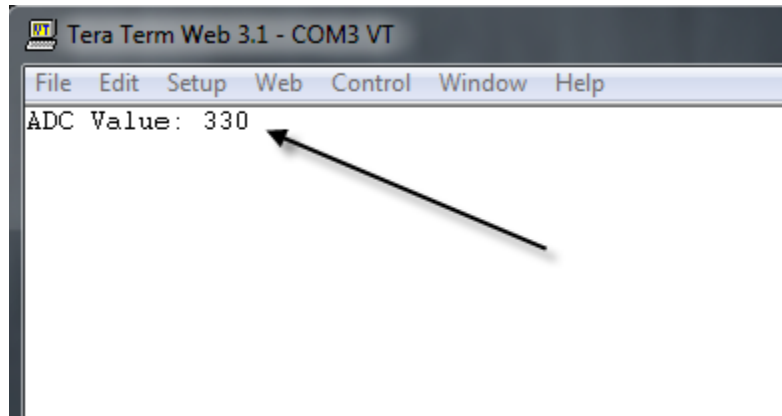


9. Entramos en Setup y después Serial Port.



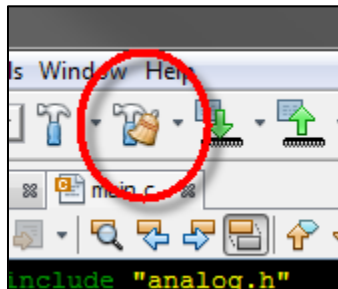


10. Deberá mostrarnos el valor de nuestro ADC en el emulador de terminal y a su vez en la LCD.

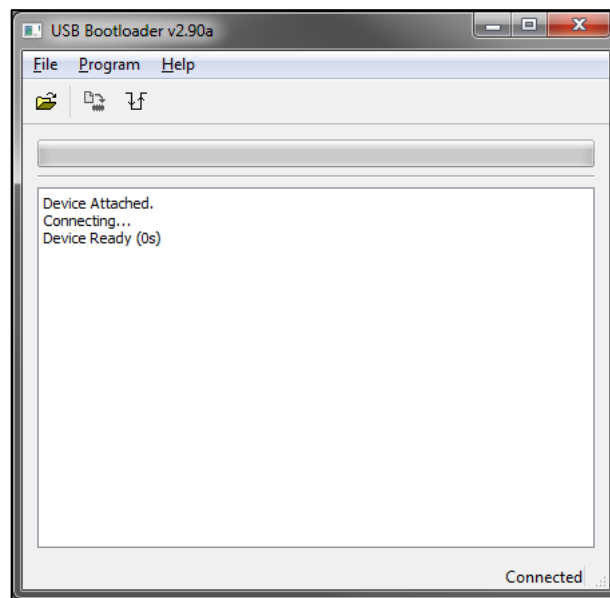
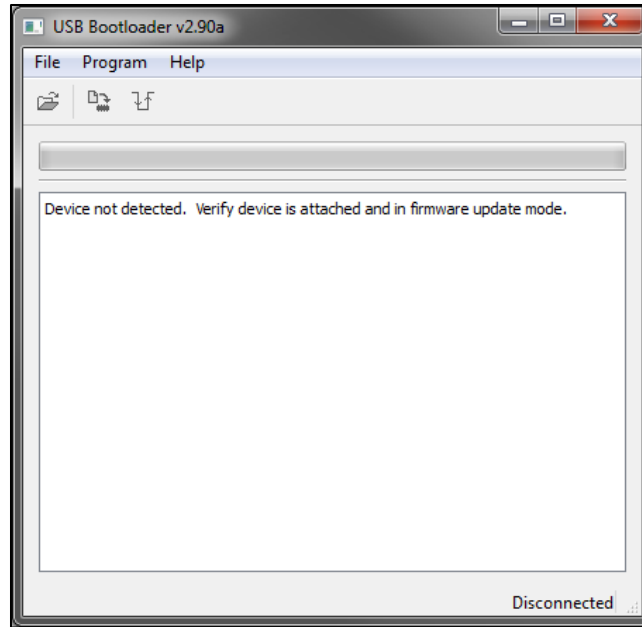


11. Compilar y programar

Al hacer clic en el ícono de compilar, y si no hay errores de compilación, el bootloader será cargado automáticamente.

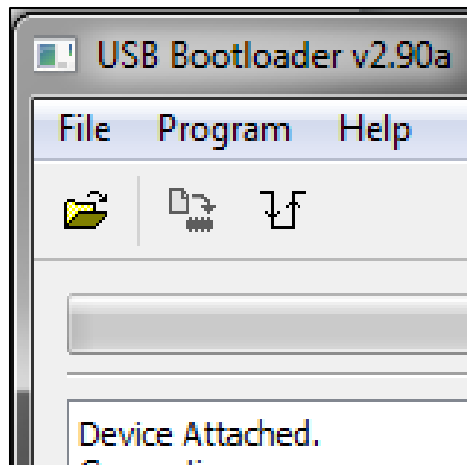


Cuando aparezca la ventana del bootloader, presione el Push-Button número 1 y conecte la fuente de voltaje o encienda el Aguijón y mantenga el PB1 presionado hasta que los LEDs empiecen a parpadear.



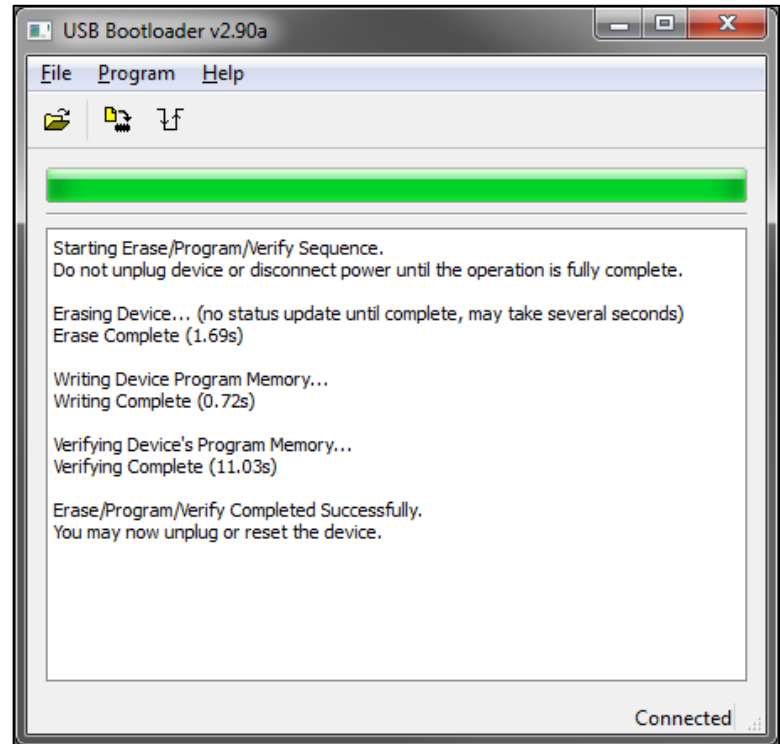
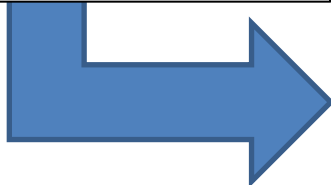
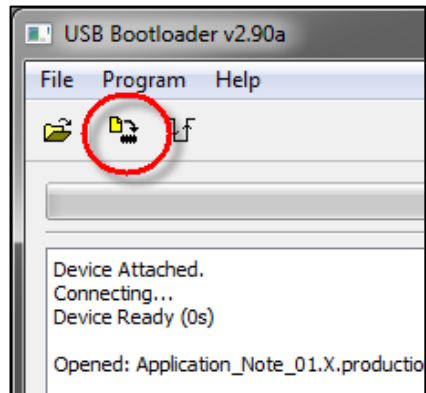
*La ventana del Bootloader indicará la conexión establecida con el aguijón:*

12. Hacer Clic en Abrir y Cargar el archivo **Application Note 18.X.production.hex**

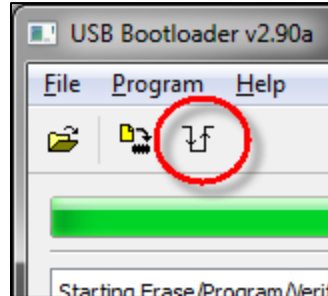


*El archivo, depende de la plataforma de hardware.*

*Una vez cargado el archivo, hacer clic en el ícono de programar.*



Hacer clic en el ícono de Reset cuando en la ventana del Bootloader se indique que se terminó de programar con éxito.



Una vez programado podemos verificar el programa corriendo en la tarjeta.

13. Para verificar el funcionamiento del programa verifique que al variar la entrada análoga introducida en el puerto ADC, esta se muestre tanto en la pantalla LCD como en el emulador de terminal.