

Manual para hacer la primera prueba con TinyDuino Basic Kit Clave: ASK1001

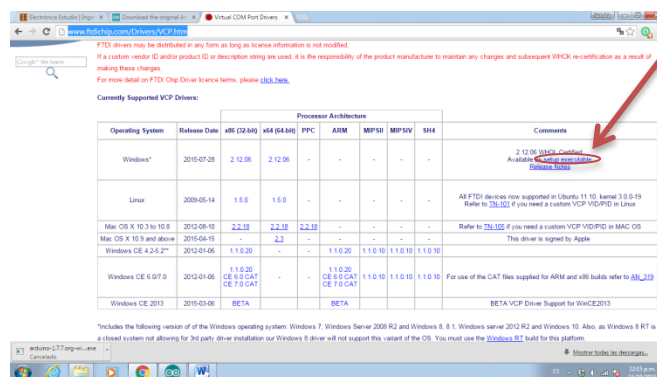
Introducción:

TinyDuino se ha convertido en uno de los favoritos para muchas personas, ya que nos ofrece capacidades en hardware y software mismas que la familia Arduino, e inclusive lo podemos operar desde el IDE de Arduino.

Este manual es un inicio para conocer a nuestro Tiny ya que les mostraré cómo configurarlo desde cargar el driver correspondiente hasta configurarlo desde la IDE Arduino con la prueba Blink.

Instalación del Driver

1. Abre tu navegador y copia la siguiente URL:
<http://www.ftdichip.com/Drivers/VCP.htm>
2. Con tu cursor ve a la parte de abajo, donde verás una tabla con los diferentes SO. En este caso utilizaré Windows 7, así que mostraré como descargar y ejecutar el driver en la parte del administrador de dispositivos. Ver la siguiente imagen:



FTDI drivers may be distributed in any form as long as license information is not modified.
If a custom vendor ID and/or product ID or description string are used, it is the responsibility of the product manufacturer to maintain any changes and subsequent WHQL re-certification as a result of making these changes.
For more detail on FTDI Chip Driver license terms, please [click here](#).

Currently Supported VCP Drivers:

Operating System	Release Date	Processor Architecture						Comments
		x86 (32-bit)	x64 (64-bit)	PPC	ARM	MIPS	SH4	
Windows*	2010-07-28	2.12.06	2.12.06	-	-	-	-	3.12.06 (VCP) Available. Click here to download.
Linux	2009-05-14	1.5.0	1.5.0	-	-	-	-	All FTDI devices now supported in Ubuntu 11.10, kernel 3.0.0-19. Refer to this link if you need a custom VCP VID/PID in Linux.
Mac OS X 10.3 to 10.8	2010-08-10	2.2.10	2.2.10	2.2.10	-	-	-	Refer to this link if you need a custom VCP VID/PID in MAC OS.
Mac OS X 10.9 and above	2015-04-15	-	2.3	-	-	-	-	This driver is signed by Apple.
Windows CE 4.2.5-2**	2012-01-06	1.1.0.20	-	-	1.1.0.20	1.1.0.10	1.1.0.10	
Windows CE 6.0/7.0	2012-01-06	1.1.0.20 CE 6.0 CAT CE 7.0 CAT	-	-	1.1.0.20 CE 6.0 CAT CE 7.0 CAT	1.1.0.10	1.1.0.10	For use of the CAT files supplied for ARM and x86 builds refer to this link .
Windows CE 2013	2015-03-05	BETA	-	-	BETA	-	-	BETA VCP Driver Support for WinCE2013

*Includes the following version of all the Windows operating system: Windows 7, Windows Server 2008 R2 and Windows 8, 8.1, Windows server 2012 R2 and Windows 10. Also, as Windows 8 RT is a closed system not allowing for 3rd party driver installation, our Windows 8 driver will not support this variant of the OS. You must use the [Windows 8.1](#) build for this platform.

**Motor tools by digicore.

Una vez descargado en nuestra PC, ve al sitio donde se descargó, generalmente se encuentra en la carpeta descargas o downloads. Una vez allí conecta tu TinyDuino.

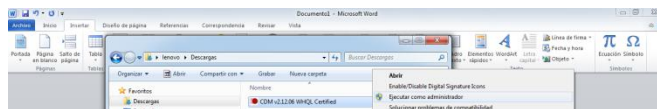
Primero va la tarjeta que trae el USB y dice ISCP, luego debajo de esta va la placa que dice TinyDuino, seguido un cable USB a V8 y a tu PC.



Nota: Cuando conectas Tiny automáticamente la barra de tareas manda un mensaje que dice “Instalando el controlador de dispositivo” cuando veas este anuncio da clic y selecciona la opción cancelar y listo.

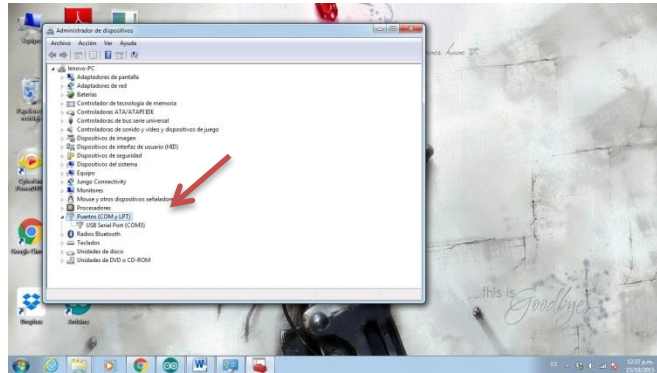
Entonces ahora ejecuta como administrador el Driver y espera a que se instale, generalmente cuando haces esto, Tiny lo detecta y lo utiliza para generar su puerto COM, que es el que te servirá para poderte comunicar desde la IDE en Arduino.

Ver la siguiente imagen:



Ok, ahora ve al administrador de dispositivos de tu PC y verifica en que puerto COM cayó Tiny, en mi caso ha caído en el COM 3.

Ver la siguiente imagen:



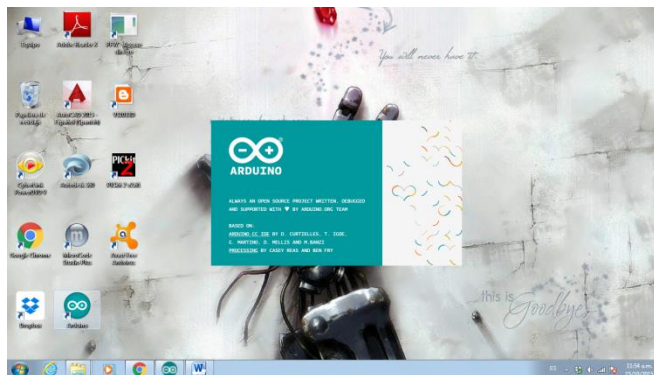
Muy bien, tu PC ya tiene un lugar para Tiny cuando vuelva a ser conectado. Aquí termina el proceso de instalación del Driver. Pasamos a la parte más divertida que es Arduino.

Ahora, por favor no desconectes tu TinyDuino para poder proseguir con la siguiente jugada.

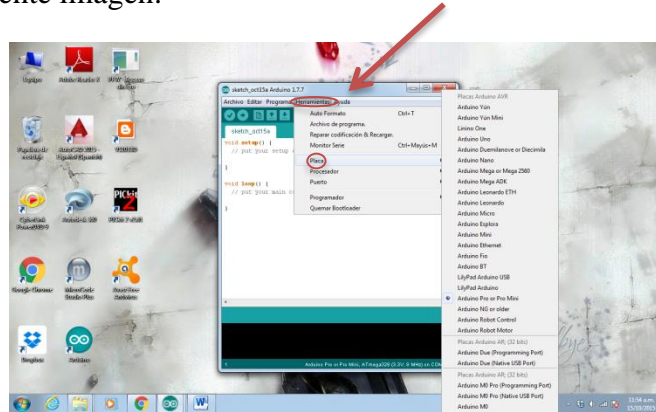
Prueba de TinyDuino ¿Estás allí?

Generalmente cuando compramos algún modulo electrónico queremos hacerle una prueba para saber que el modulo está trabajando bien y así saber en cuanto le hagamos una prueba o programa más complejo lo ejecute con éxito y si no, querrá decir que algo estamos haciendo mal.

1. Ejecutar Arduino. Vea la siguiente imagen:

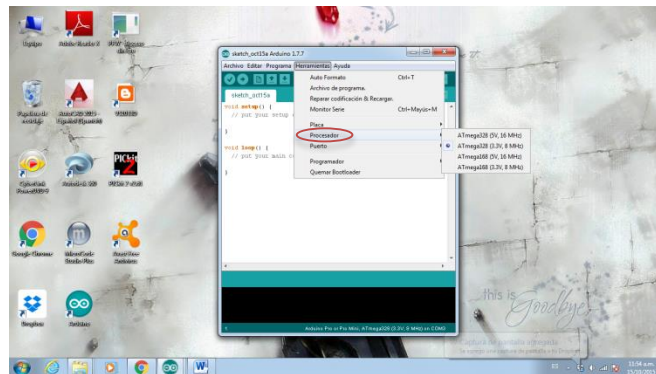


2. Aparecerá la pantalla del sketch. Ve a herramientas para configurar tu TinyDuino. Vea la siguiente imagen:

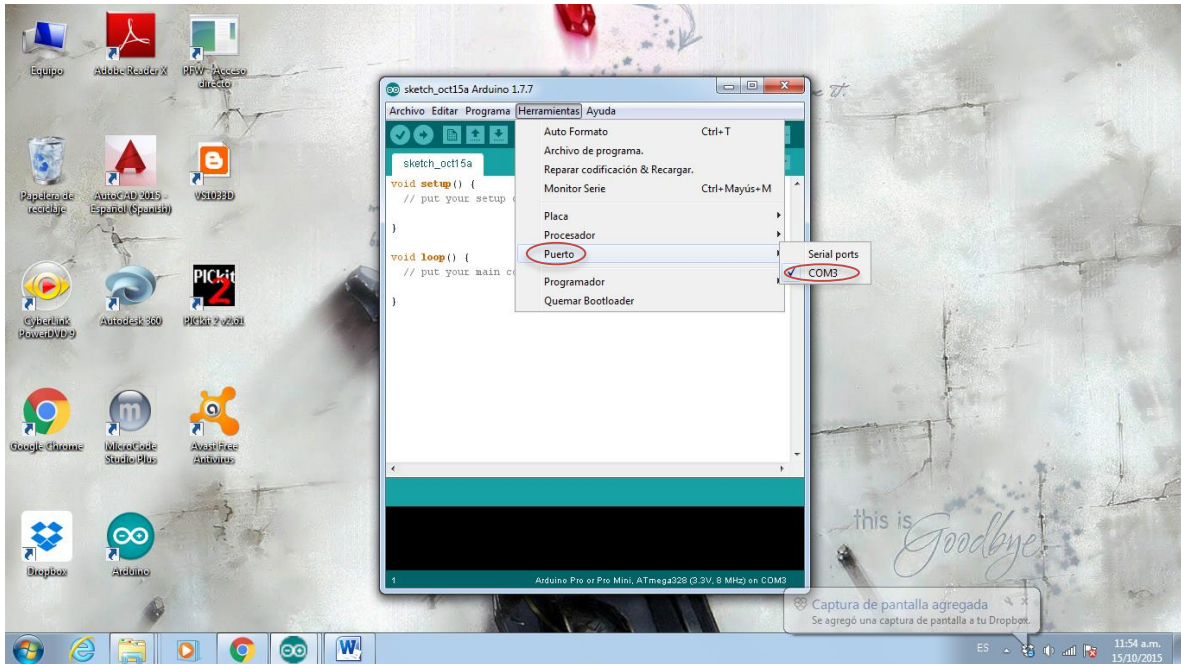


En el paso 2 es importante que selecciones el tipo de placa, aunque no sea Tiny “puesto que no aparecerá esta opción” Selecciona Arduino Pro or Pro Mini

3. Ahora selecciona el procesador, en este caso será el que trae nuestro Tiny. Ver la siguiente imagen:

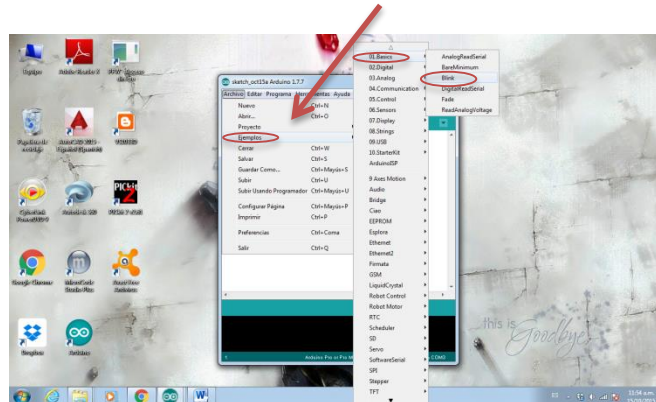


4. Por último y más importante el COM, si no lo seleccionas correctamente será imposible comunicarnos con nuestra placa. Ver la siguiente imagen:



¡Perfecto! Ya estás listo para cargarle el Blink.

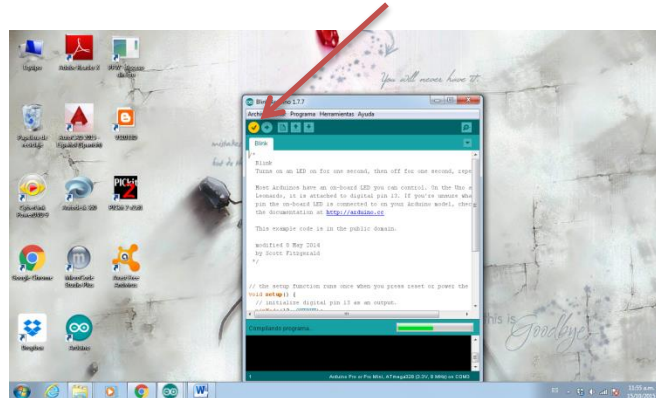
5. Ve a la parte de Archivo → Ejemplos → 01. Basics → Blink
Vea la siguiente imagen:



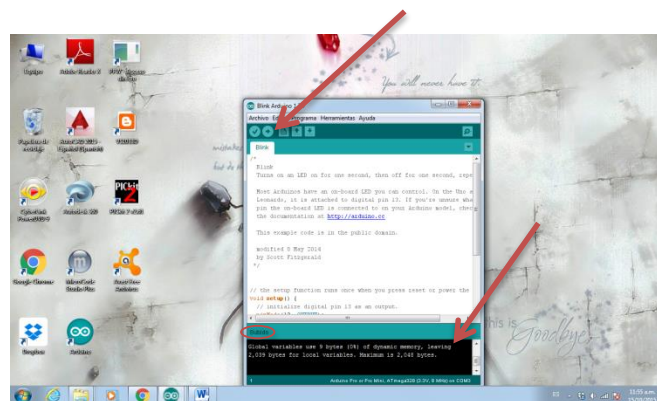
De acuerdo, ya solo falta cargar el programa

6. Lo primero que deberás hacer será, compilar o verificar, esto para saber que no haya errores en el programa, de lo contrario no lo podrá leer tu tarjeta. Da clic en la paloma.

Ver la siguiente imagen:



7. ¡Cargar a TinyDuino!
Da clic al icono de la flecha →
Vea la siguiente imagen:



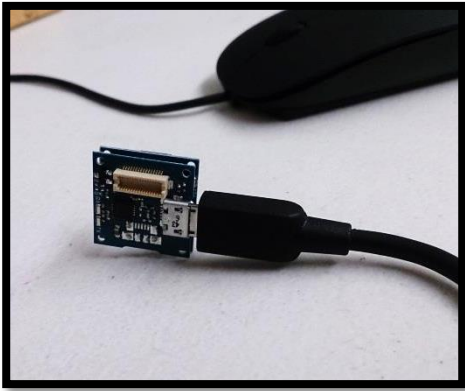
8. Bien no han aparecido errores, esto quiere decir que TinyDuino está listo y seguramente ya hayas notado que tu plaquita destella un LED color verde con la velocidad de 1 segundo al encenderse y 1 segundo al apagarse.

Recuerda que para configurar los tiempos de pausa en programación C, la mayoría de las veces este valor se da en ms (milisegundos).

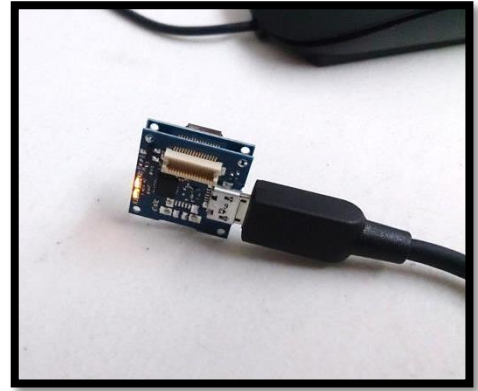
Ahora te muestro como debe funcionar tu placa ya programada con el Blink.

Cuando cargaste el programa al dar clic a la flecha. Tiny debió haber destellado un par de leds que trae el convertidor USB. Esto indica que el convertidor está trabajando bien y que está transfiriendo la información a Tiny.

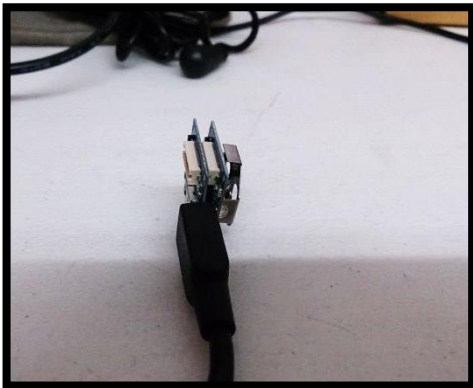
El LED destellará 30 ms/s
aprox.



LED 1000 ms apagado



LED 1000 ms encendido



Con esta prueba tan sencilla, hemos concluido, podrás proseguir si tú lo deseas, puede ser modificando los tiempos en el programa Blink o bien aplicando Tiny en un proyecto más complejo.

El equipo de Electrónica Estudio espera haberte ayudado