

EUSKALHACK
SECURITY
TALKS

Playing with USB devices for fun



euskaltel 
FUNDAZIOA



@EMSOL



SERGIO & ROBER

“Playing with USB Devices for Fun”



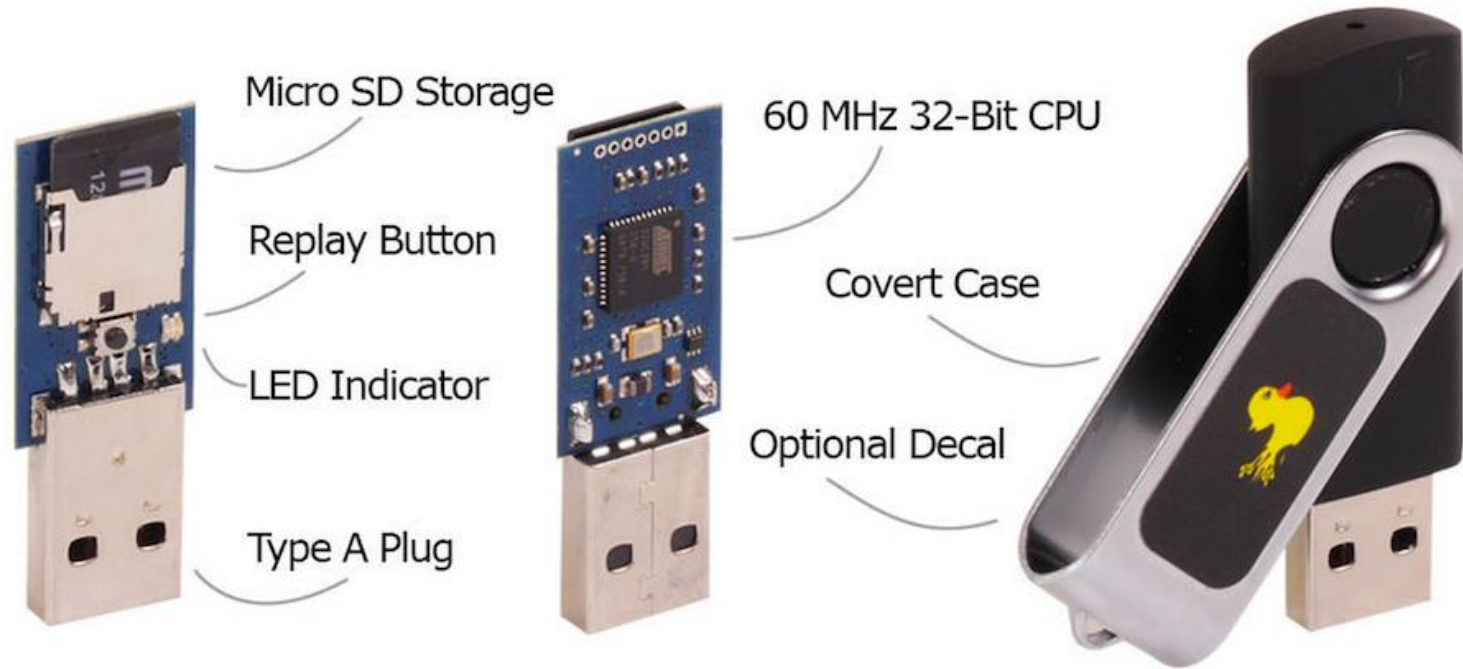
@euskalhack

“Playing with USB Devices for Fun”

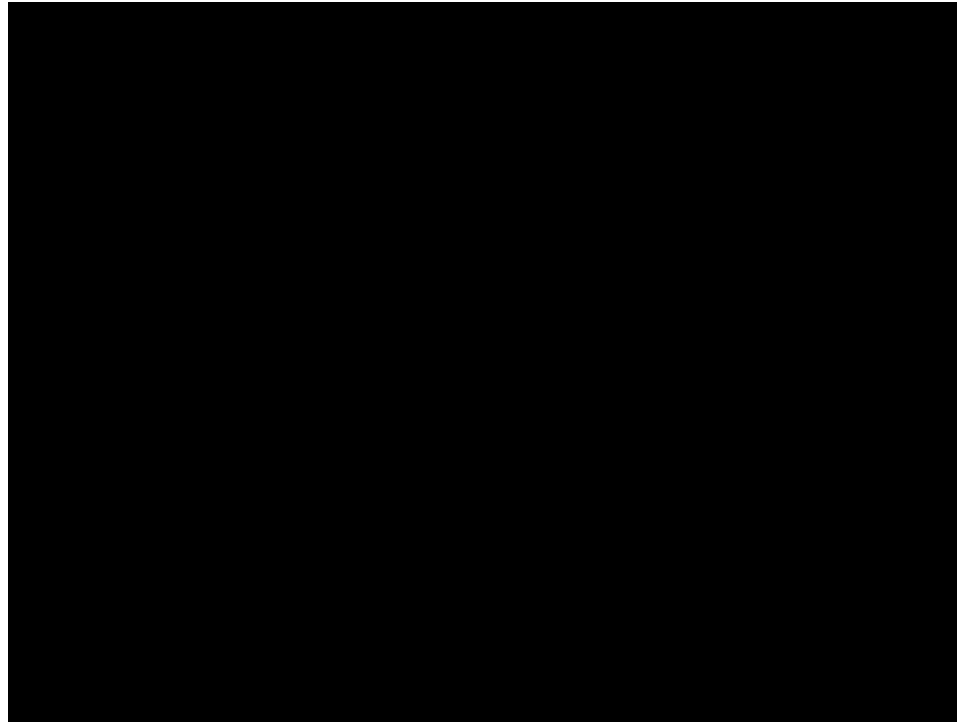


USB
UNIVERSAL SERIAL BUS

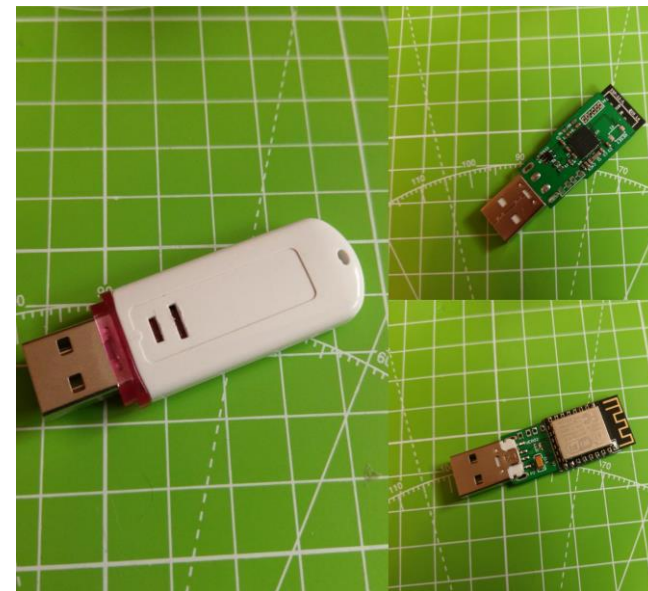
“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”



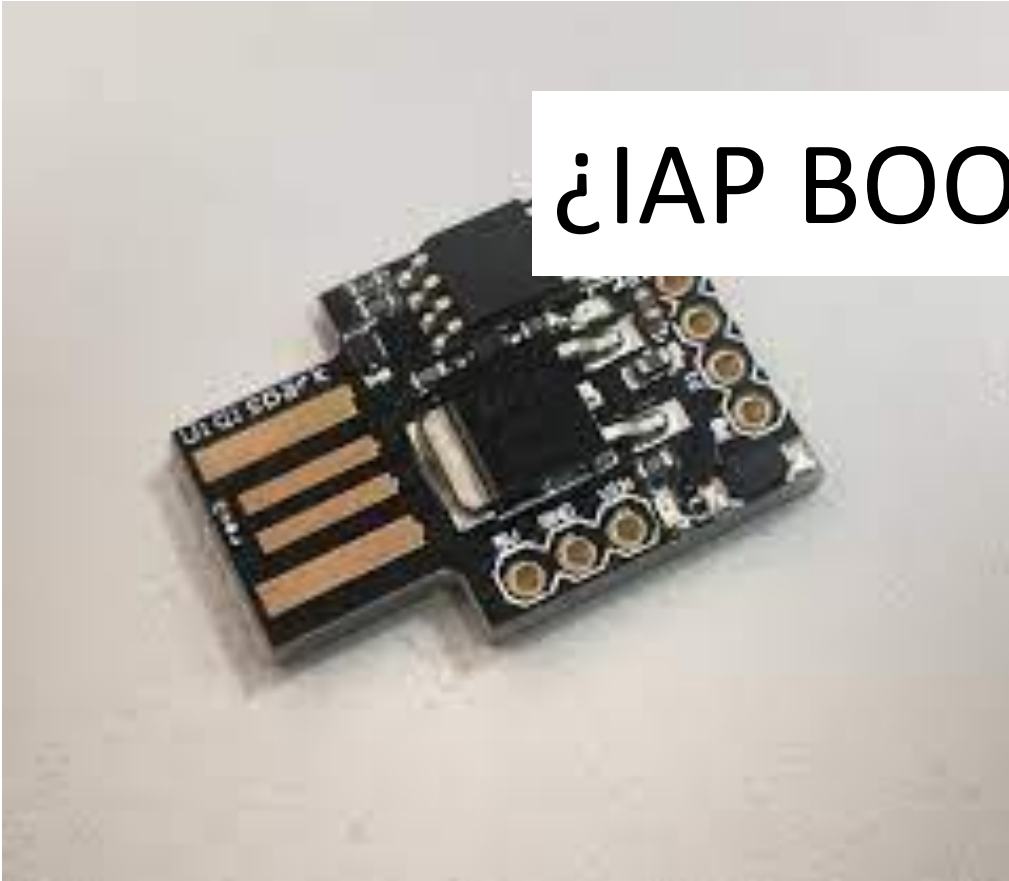
“Playing with USB Devices for Fun”



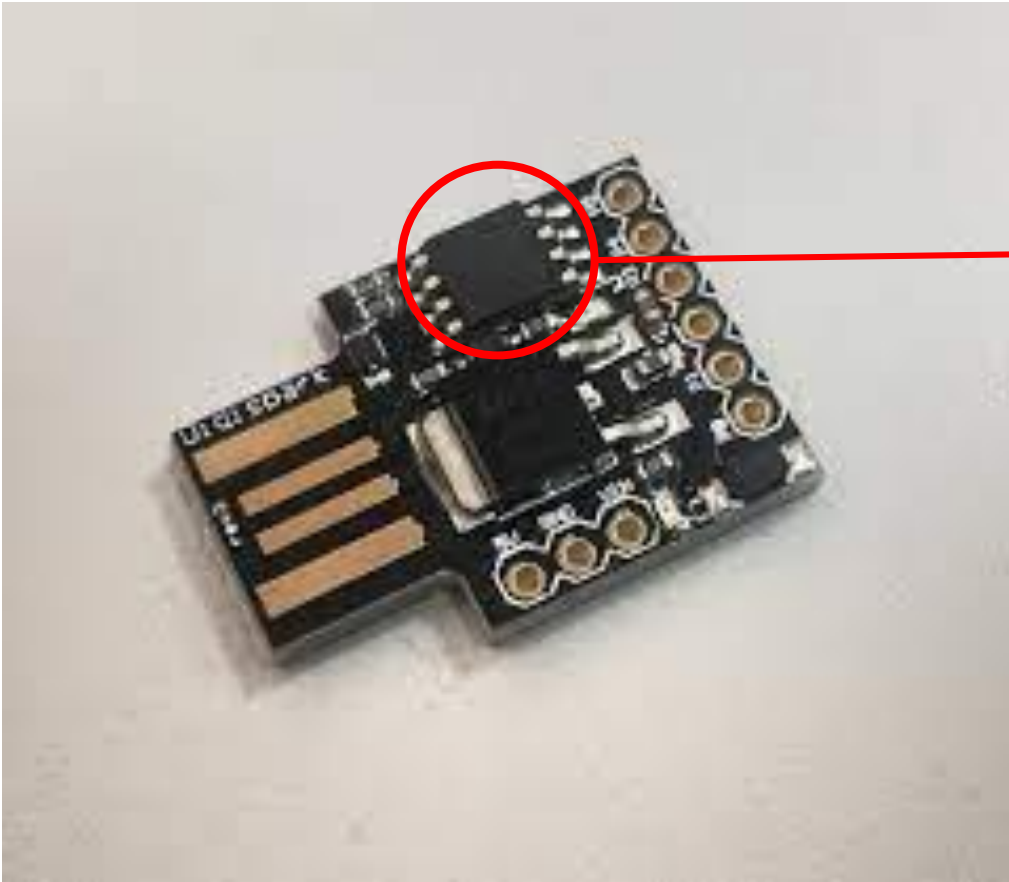
“Playing with USB Devices for Fun”

PROGRAMABLE CON ARDUINO IDE

¿IAP BOOTLOADER?



“Playing with USB Devices for Fun”



8 bits-CPU

16'5Mhz (para que se comuniquen mediante USB)

512 bytes SRAM

8K Flash

512 bytes EEPROM

“Playing with USB Devices for Fun”

¿Por qué Attiny85?



-Es un Microcontrolador Sencillo y bajo coste pero con capacidades USB (Is) gracias a la libreria V-USB.

-Apenas necesita circuiteria auxiliar.

- Gracias al bootloader micronucleus podemos reprogramarlo sin circuitería externa/auxiliar.

“Playing with USB Devices for Fun”

OBJETIVO 0 - USB

“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”



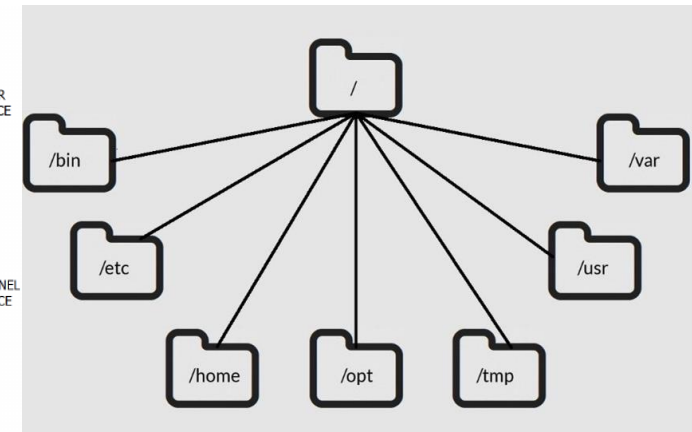
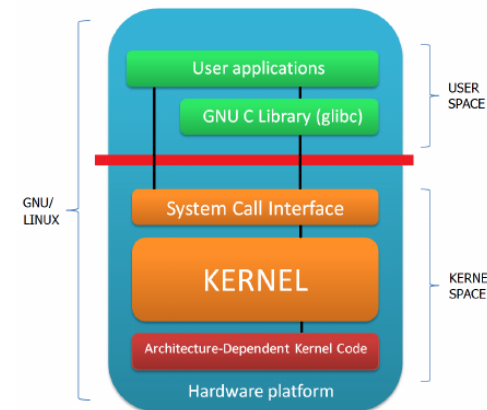
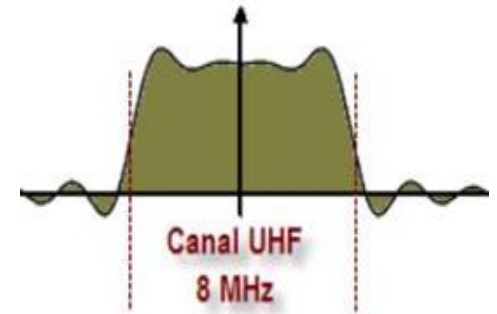
TCP/IP Model

Network Access Layer

Internet Layer

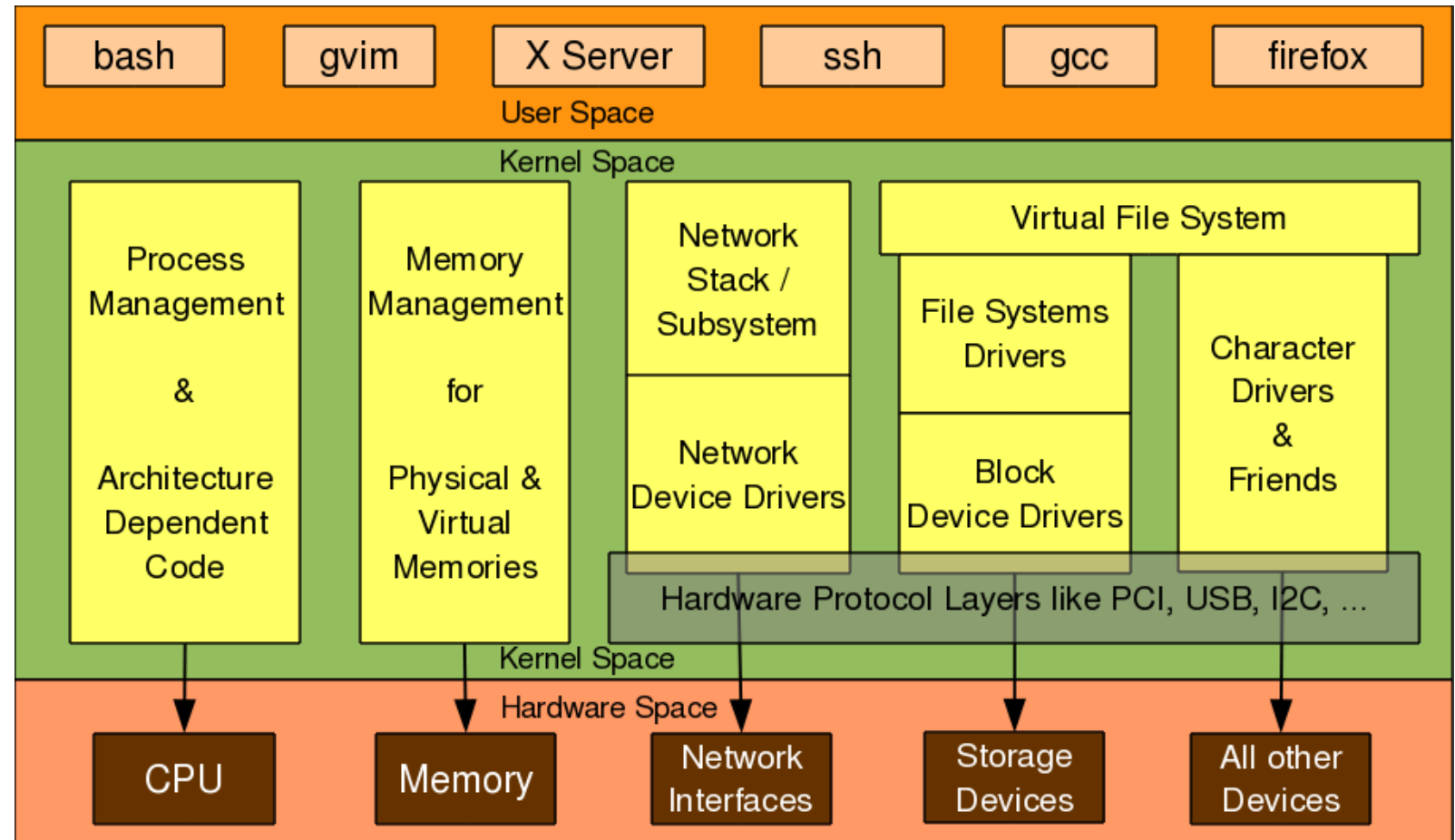
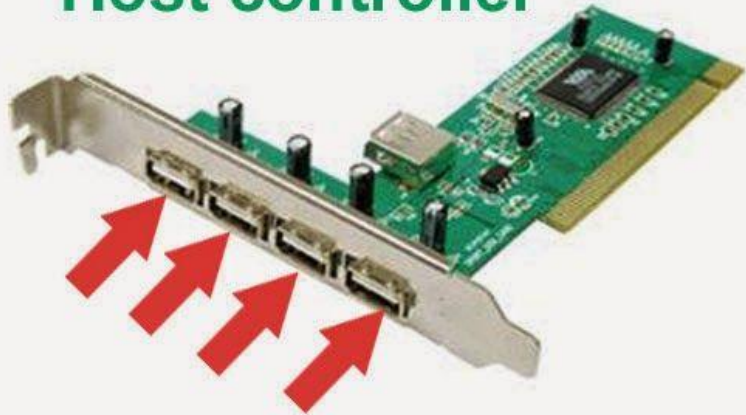
Transport layer

Application Layer

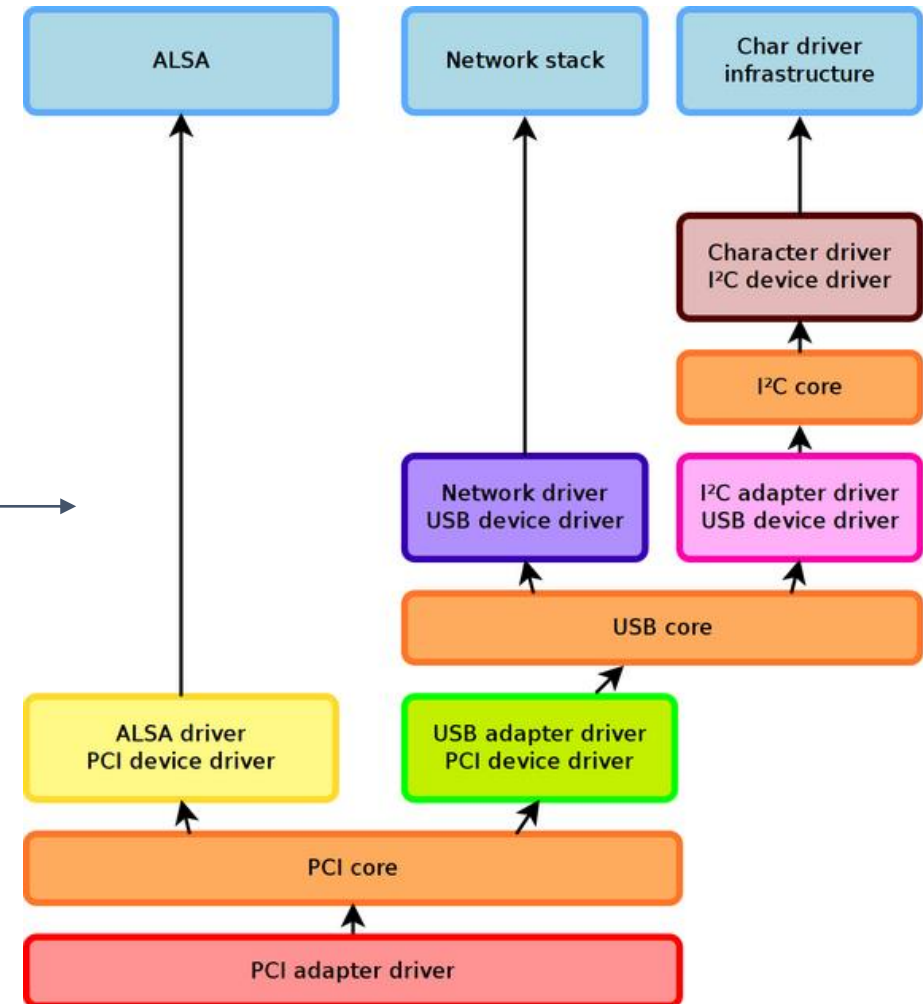
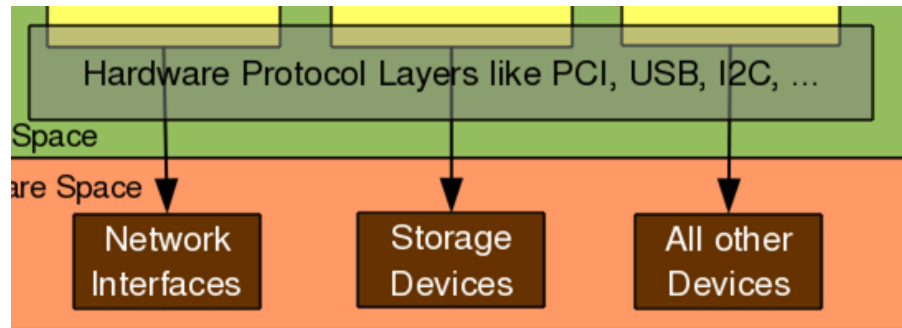


“Playing with USB Devices for Fun”

Hi speed USB
Host controller



“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”

LOW SPEED

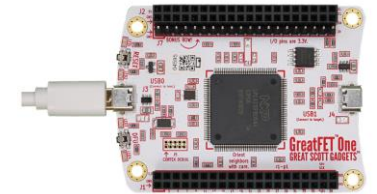
1.5 Mbit/s

FULL SPEED

12 Mbit/s

HIGH SPEED

480 Mbit/s



SUPER SPEED

5.0 Gbit/s

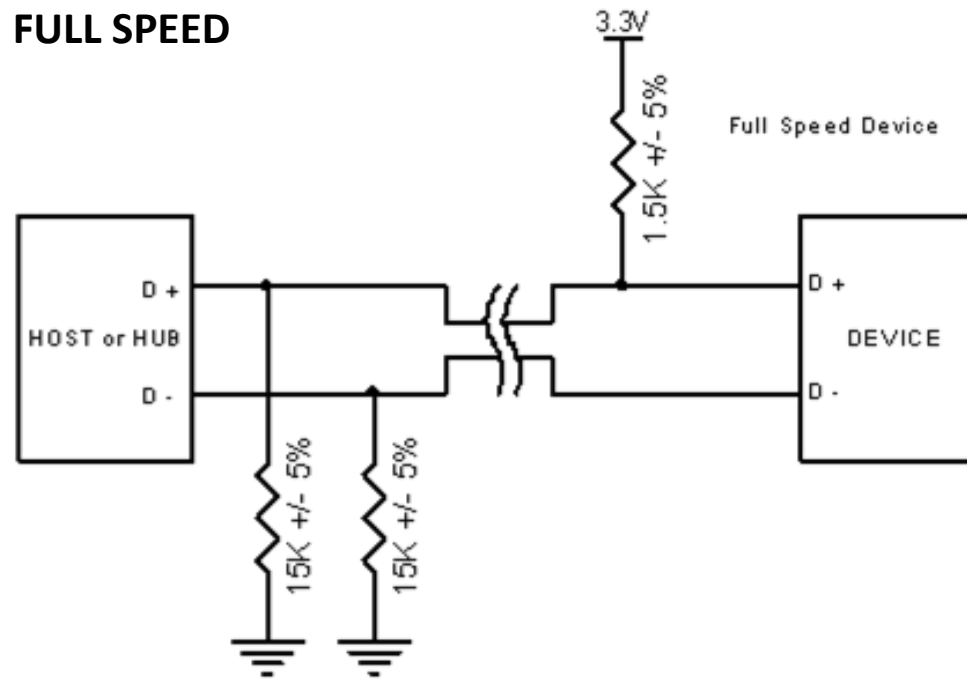
SUPER SPEED PLUS

10 Gbit/s

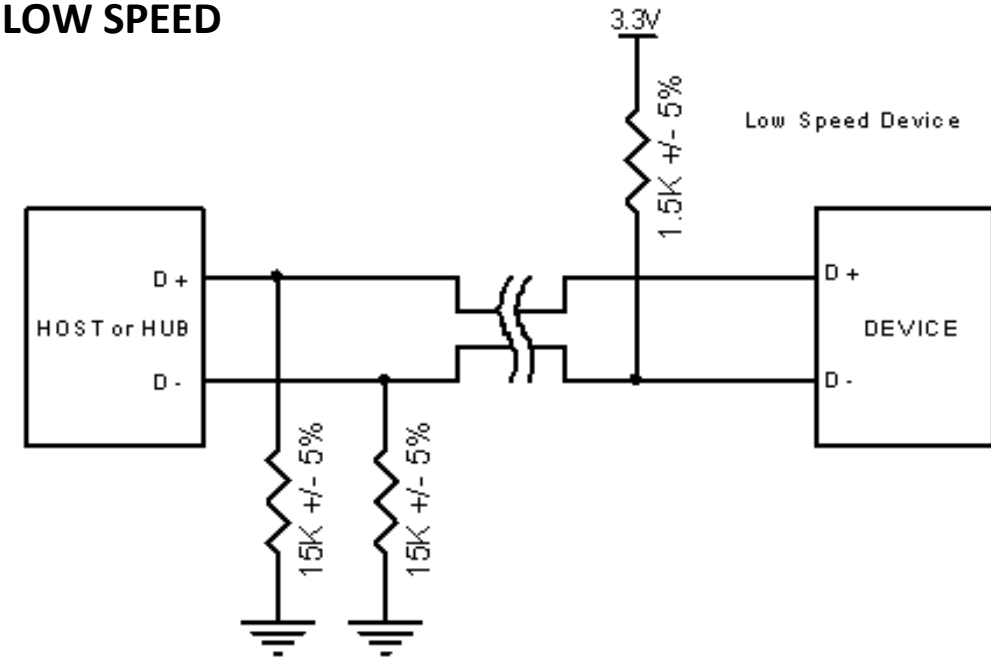
“Playing with USB Devices for Fun”



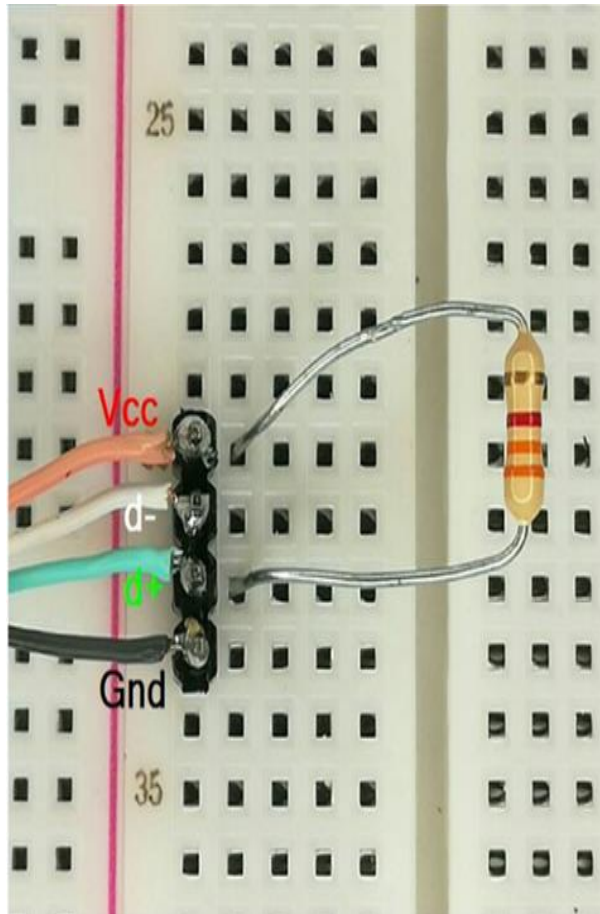
FULL SPEED



LOW SPEED



“Playing with USB Devices for Fun”



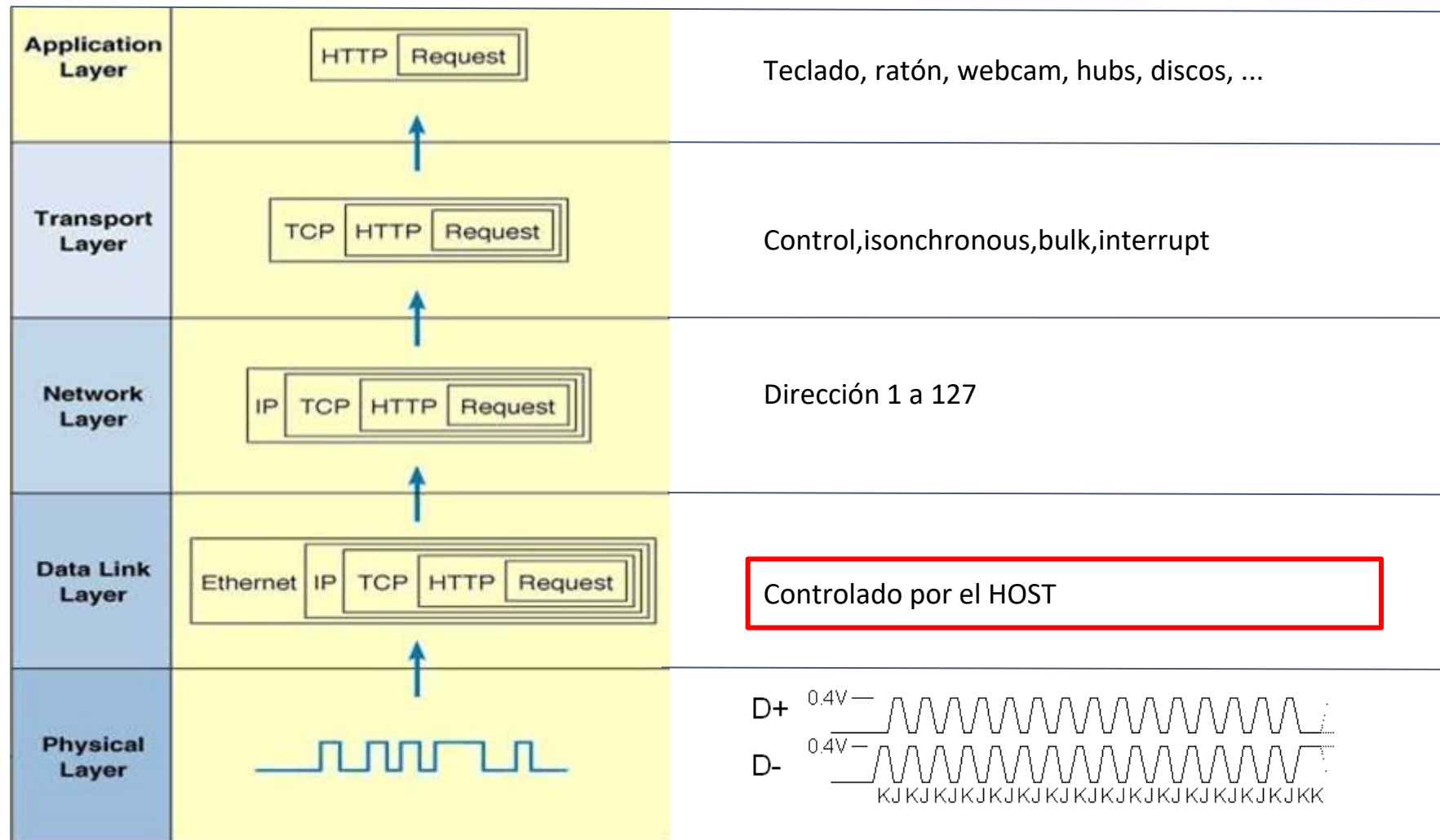
```
----- Connection Information -----  
Connection Index      : 0x01 (Port 1)  
Connection Status    : 0x02 (DeviceFailedEnumeration)  
Current Config Value  : 0x00 (Configuration 0)  
Device Address        : 0x00 (0)  
Is Hub                : 0x00 (no)  
Device Bus Speed      : 0x01 (Full-Speed)  
Number Of Open Pipes  : 0x00 (0 pipes to data endpoints)
```

“Playing with USB Devices for Fun”

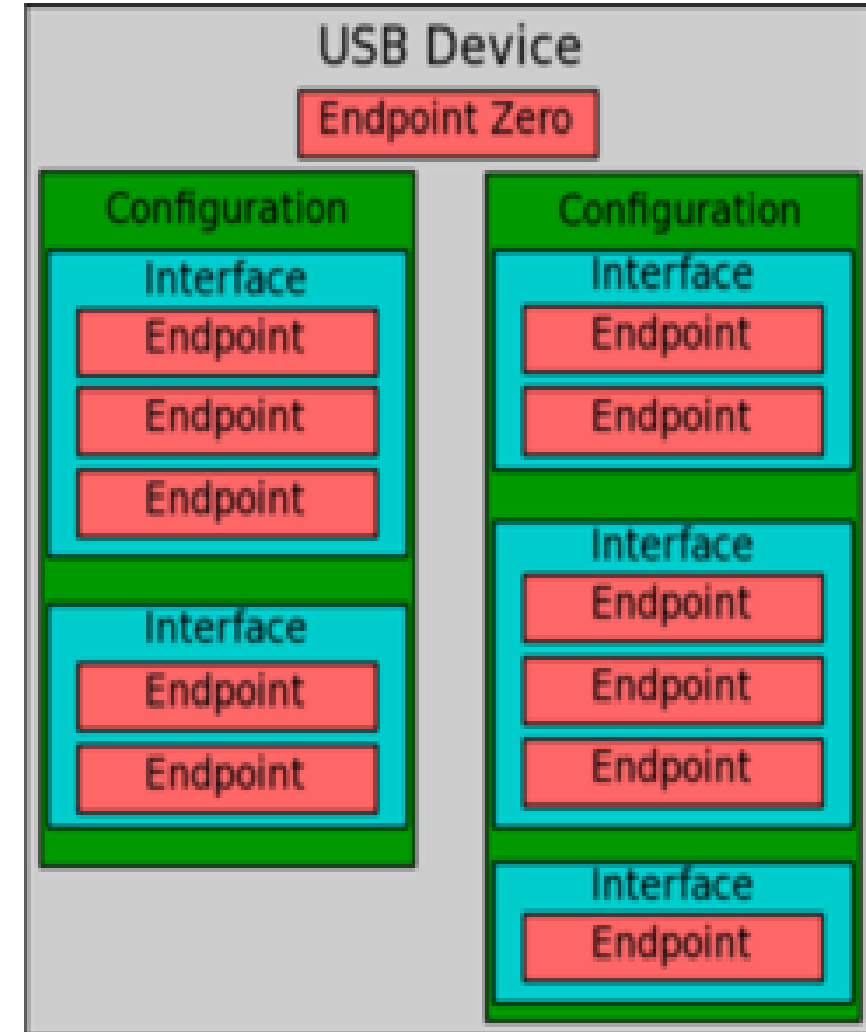
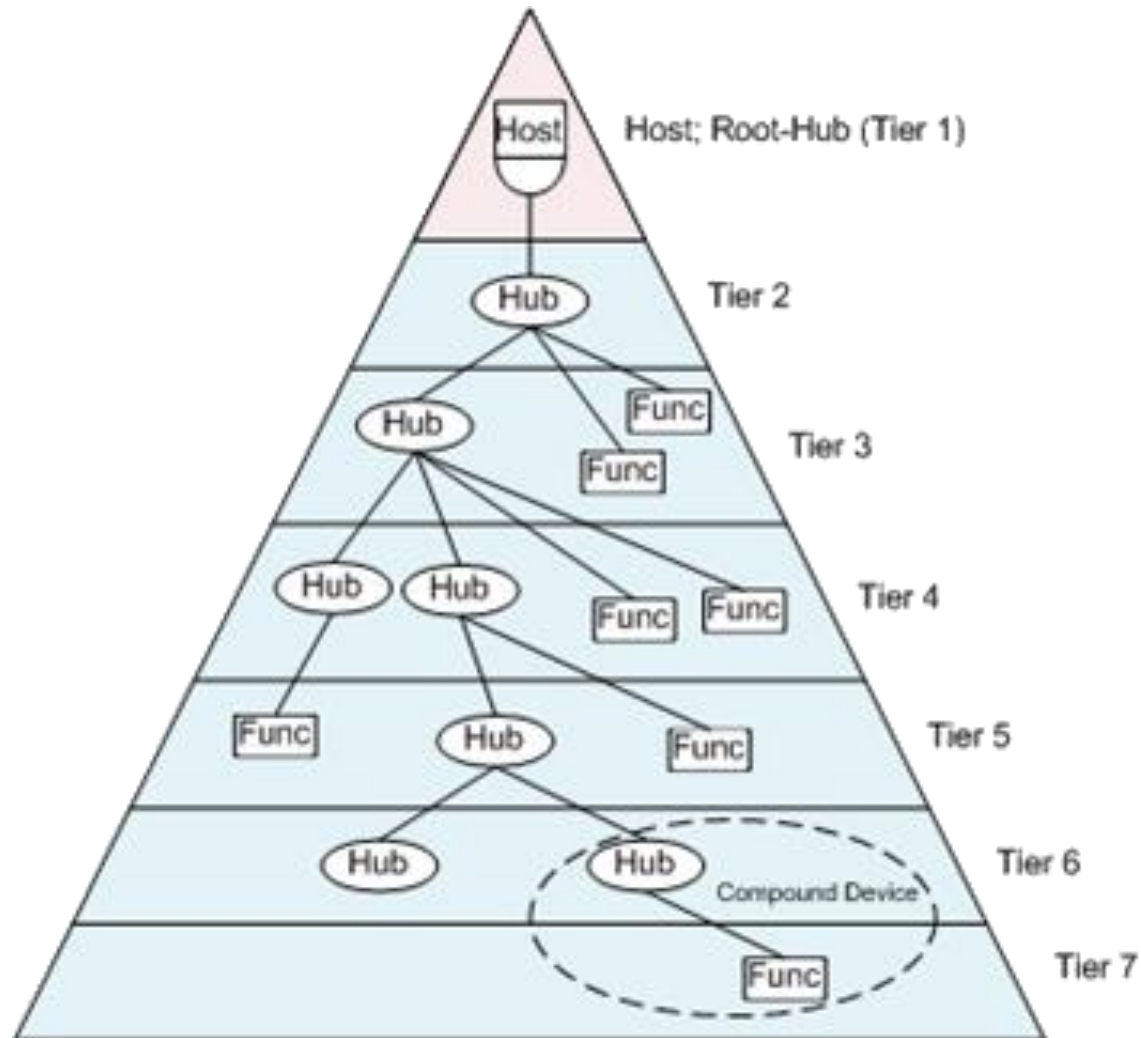
```
===== USB Device =====

+++++ Device Information +++++
Device Description      : Dispositivo USB desconocido (Error de solicitud de descriptor de dispositivo)
Device ID              : USB\VID_0000&PID_0002\6&7323E35&0&1
Hardware IDs           : USB\DEVICE_DESCRIPTOR_FAILURE
Driver KeyName         : {36fc9e60-c465-11cf-8056-444553540000}\0059 (GUID_DEVCLASS_USB)
Driver Inf             : C:\WINDOWS\inf\usb.inf
Legacy BusType         : PNPBus
Class                  : USB
Class GUID             : {36fc9e60-c465-11cf-8056-444553540000} (GUID_DEVCLASS_USB)
Enumerator             : USB
PDO                   : \Device\USBPDO-6
Location Info          : Port_#0001.Hub_#0002
Manufacturer Info      : (Controladora de host USB estándar)
Capabilities           : 0x64 (Removable, SilentInstall, RawDeviceOK)
Status                 : 0x01806400 (DN_HAS_PROBLEM, DN_DISABLEABLE, DN_REMOVABLE, DN_NT_ENUMERATOR, DN
Problem Code           : 43 (CM_PROB_FAILED_POST_START)
Address                : 1
HcDisableSelectiveSuspend : 0
EnableSelectiveSuspend  : 0
SelectiveSuspendEnabled : 0
EnhancedPowerMgmtEnabled : 0
IdleInWorkingState     : 0
WakeFromSleepState     : 0
Power State            : D3 (supported: D0, D2, D3, wake from D0, wake from D2)
```

“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”

Conectar el dispositivo

Detectar la conexión

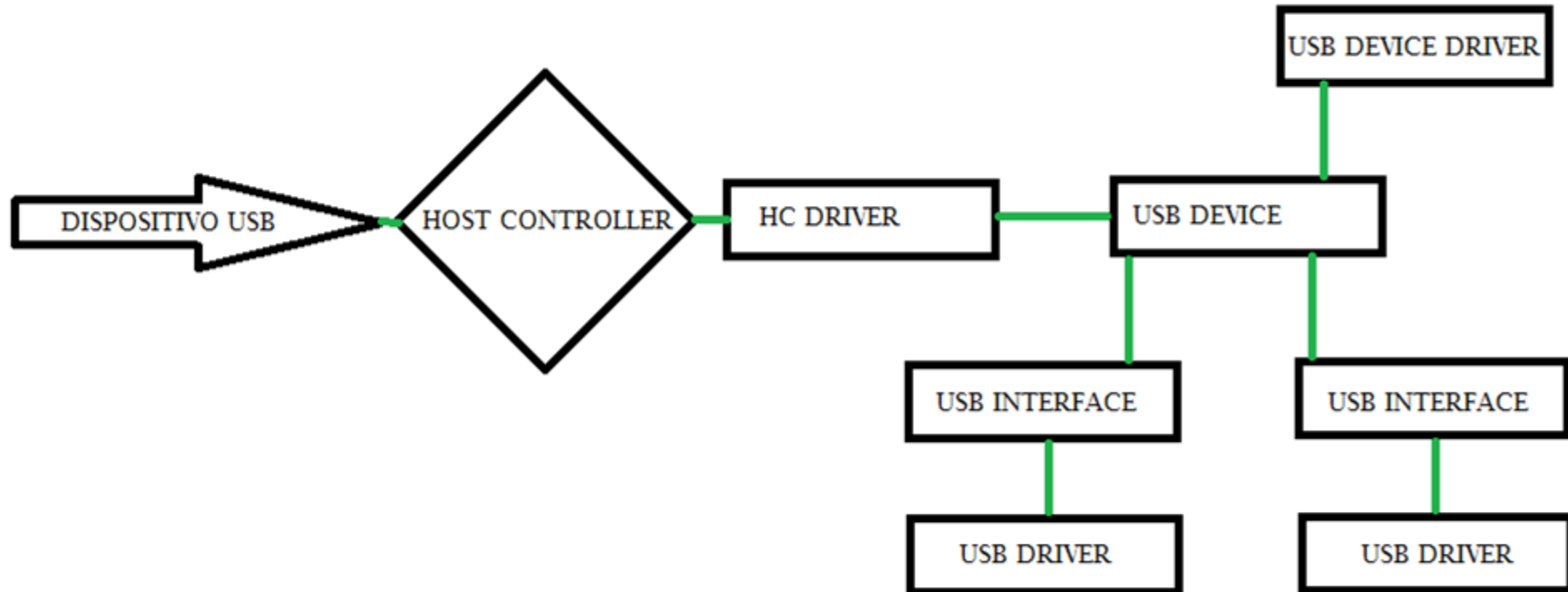
Asignar dirección

Recoger la información del dispositivo

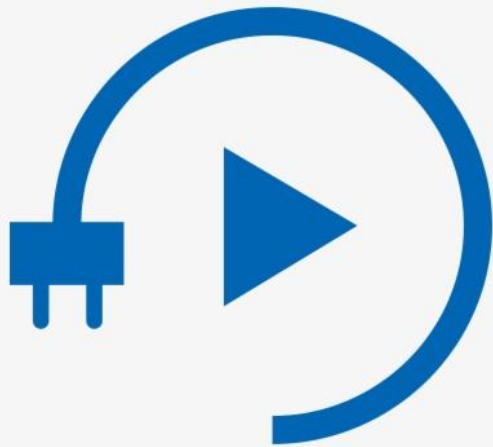
Elegir la configuración

Elegir los drivers para las interfaces

“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”

Las resistencias [pull-up | pull-down] son resistencias utilizadas en circuitos lógicos para asegurar un nivel lógico bien definido en un pin bajo cualquier condición => 1 | 0 por tanto sirven para asegurar un valor por defecto en una patilla cuando no está conectada a nada.

En electrónica digital, la lógica triestado permite puertos de salida con valor 0, 1 o alta impedancia (Hi-Z del inglés High Impedance).

La utilidad del tercer estado (Hi-Z) es borrar la influencia de un dispositivo del resto del circuito, osea cuando ponemos una patilla en alta impedancia lo que hay detrás desaparece a nivel electrónico.

Como poner una patilla del attiny en alta impedancia

Es Tan sencillo como configurar la patilla como entrada en lugar de salida Y escribir un cero/Low para desactivar el **pull-up interno** típico de las patillas I/O

```
pinMode (USB_PullUp, INPUT) ;      //Pongo Pin en Alta Impedancia  
digitalWrite (USB_PullUp, LOW) ;    //desactivo PullUp
```


“Playing with USB Devices for Fun”

OBJETIVOS DE NUESTRO DISPOSITIVO

Vamos a “jugar” con el Attiny hasta conseguir un circuito lo suficientemente compacto para introducirlo dentro de un cable USB

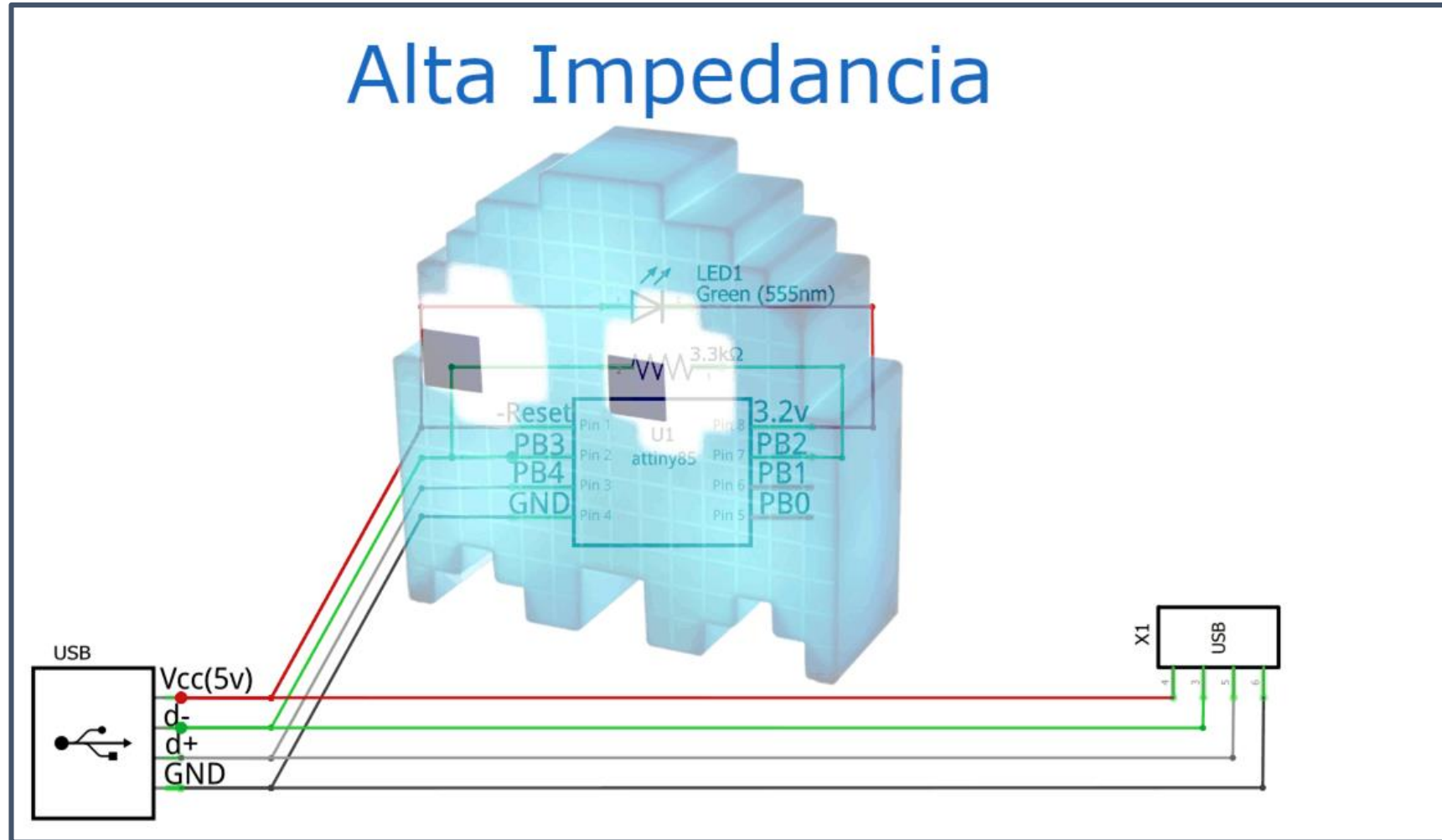
Barato 4 euros en amazon

Que sea fácil de conseguir los componentes

Que sea fácil de montar

Además de actuar como teclado el dispositivo final (móvil) que conectemos al final del cable tiene que ser funcional

“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”

objetivo de desarrollo 1:

Conseguir comunicación USB del Attiny85 lo más compacta posible , además queremos ocultar el bootloader moviendo la resistencia y además la resistencia tiene que servir para indicar conexión pullup d-

Objetivo de desarrollo 2:

Triestado alta impedancia oculta el Attiny85 y permite comunicación USB dispositivo final host mediante el mismo cable

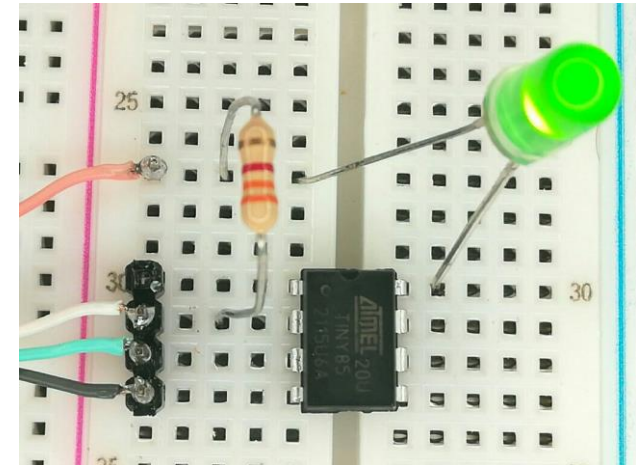
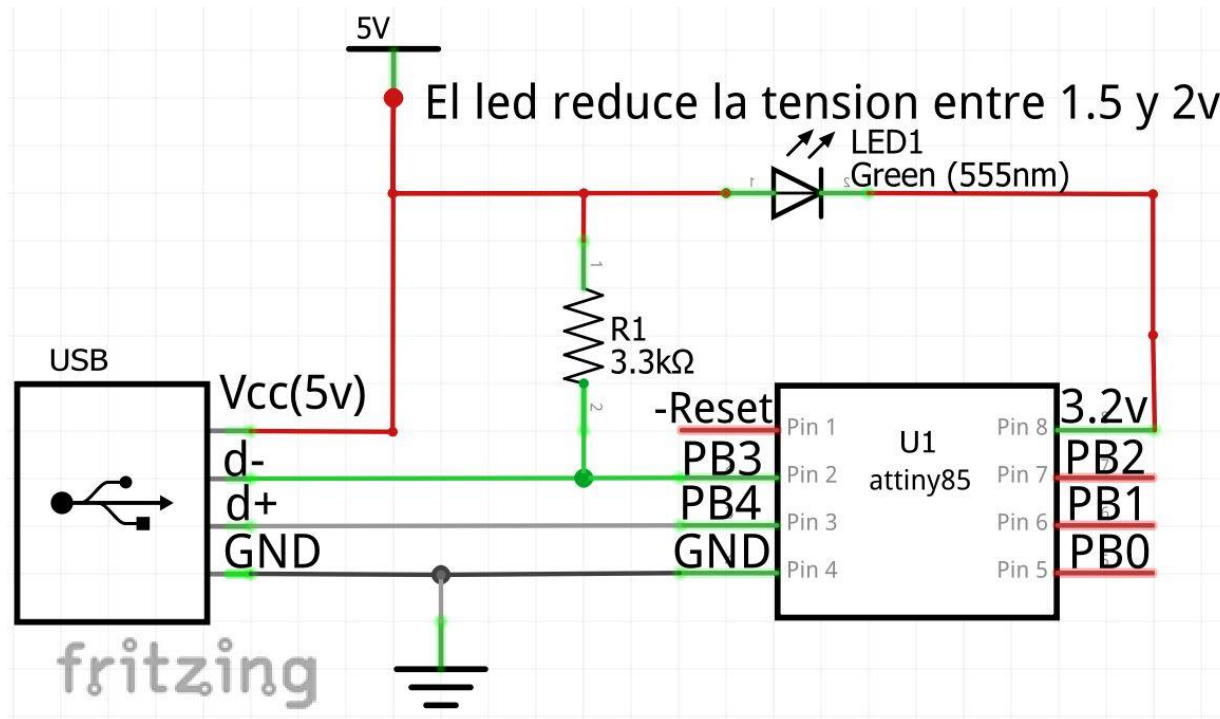
Objetivo de desarrollo 3:

Detectar cuándo lanzar el payload el Attiny85 se presenta como usb lowspeed HID y lanza la carga.

“Playing with USB Devices for Fun”

OBJETIVO 1

“Playing with USB Devices for Fun”



Nota: El led no es un regulador de tensión ni un Zener.
Pero nos servirá si el circuito consume menos de 15mA (Intensidad máxima de un led)
Reduciendo la tensión entre 1.5 y 2v (dependiendo del modelo de led (color))

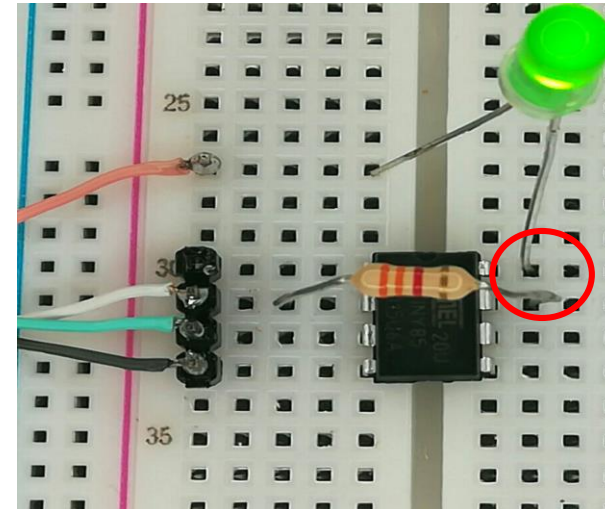
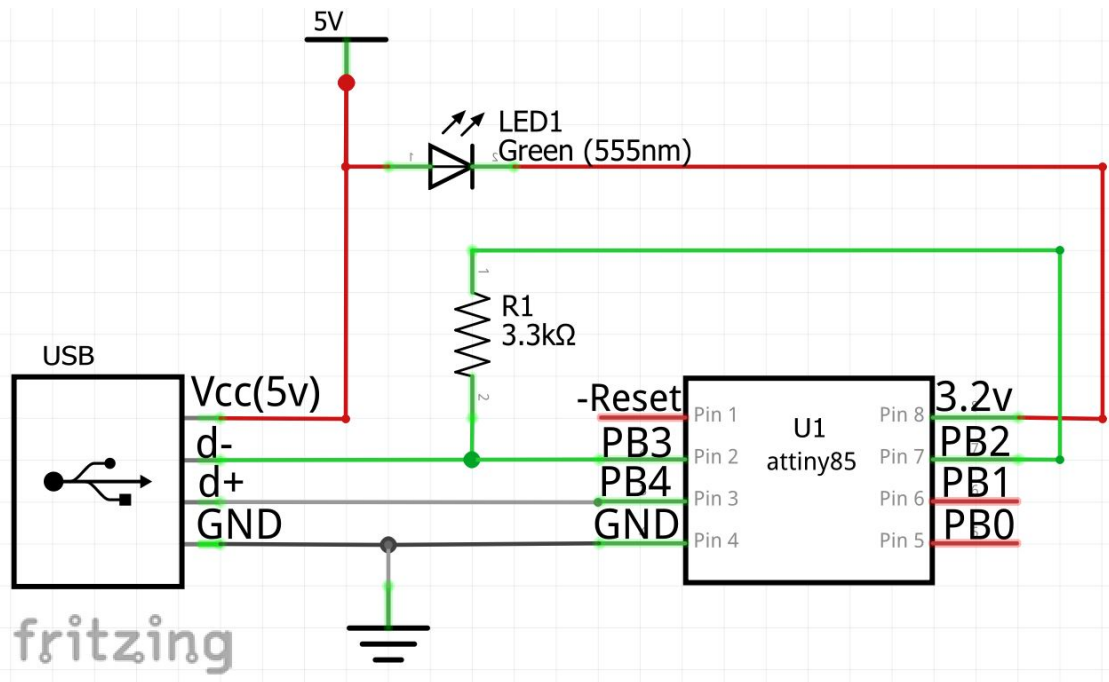
“Playing with USB Devices for Fun”

OBJETIVO 2

“Playing with USB Devices for Fun”

Pongo resistencia en PB2 en lugar de 3.2v

Así puedo activar y desactivar la resistencia Pull-Up , ocultar el bootloader y controlar la detección del dispositivo via USB



Así consigo “desconectar” / “conectar” el USB electrónicamente.

Si pongo PB2=1 es como si conectara la resistencia Pull-Up

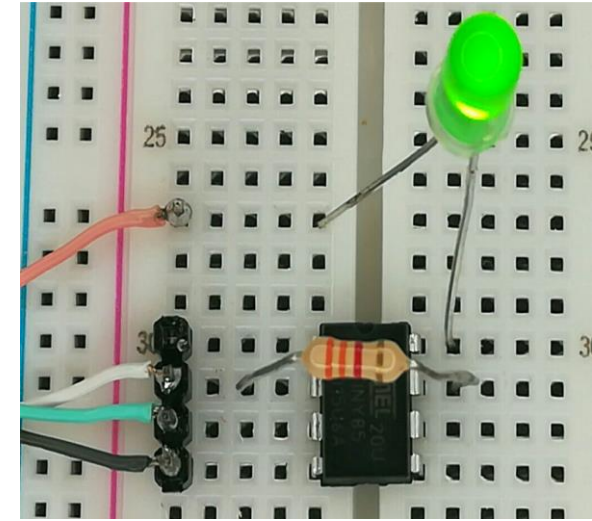
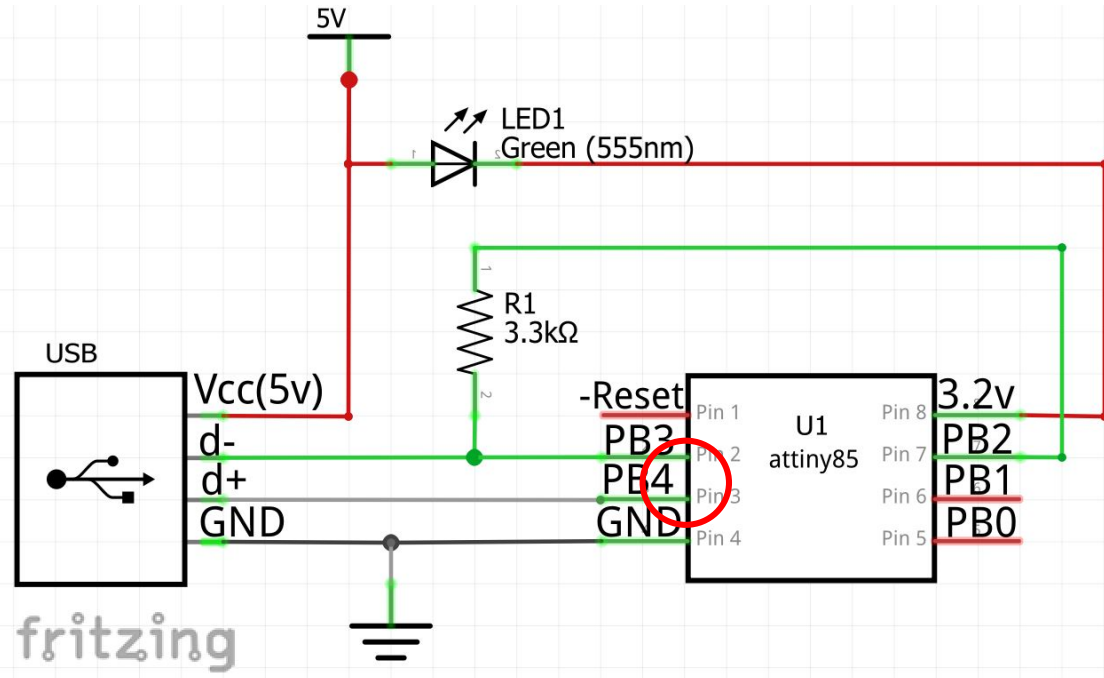
Si pongo PB2 en alta impedancia/triestado es como si desconectara la resistencia Pull-Up

Nota: al hacer esto “perdemos el BootLoader” pero esto puede ser una ventaja.....

“Playing with USB Devices for Fun”

OBJETIVO 3

“Playing with USB Devices for Fun”

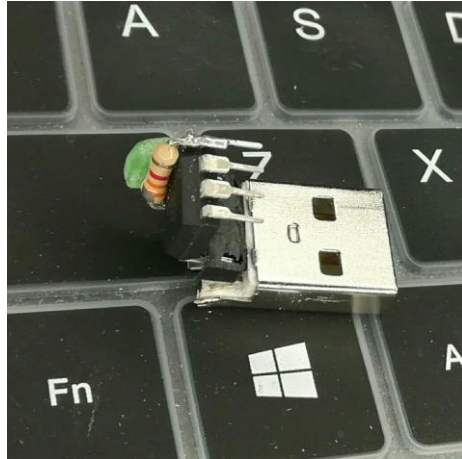
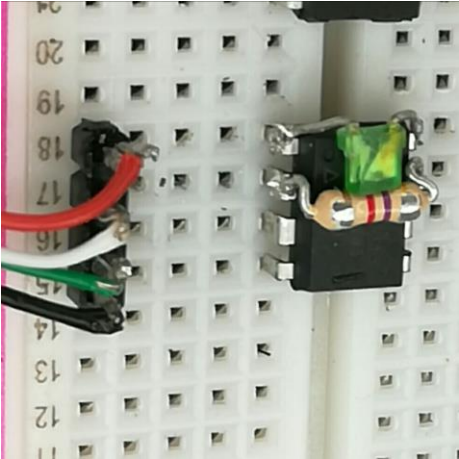


Mientras el dispositivo mantiene las patillas PB2, PB3 y PB4 en alta impedancia. (está “oculto” a nivel electrónico) Puedo monitorear las patillas PB3 y PB4 (conectadas a d- y a d+). Así consigo detectar si hay otro dispositivo usb conectado al mismo cable.

En nuestro caso esperamos a que se conecte un dispositivo al cable USB (ej un móvil para cargar o pasar fotos) y luego volvemos a esperar a que se desconecte de nuevo dicho dispositivo,

A los 10 segundos de la desconexión el attinity se presenta como usb lowspeed HID y lanza la carga.

“Playing with USB Devices for Fun”



“Playing with USB Devices for Fun”

