

## Requirements

- Pentium III 600MHz
- 128 MB RAM
- USB 2.0 host controller (recommended)
- Resolution 800x600 256 colours
- Windows® 2000 Service Pack 4,  
Windows® XP Service Pack 1 or higher
- Internet Explorer 5.0 or higher

## Specifications

### Firmware / Upgrade

- Firmware is automatically updated with each software upgrade

### Package

- Dimensions: 85 x 80 x 45 mm
- Weight: 100 g

### Analyzer connector (host computer)

- USB 2.0 high-speed (480 Mbit/s)

### DUT connectors (bus to be analysed)

- USB 1.1 low-speed (1.5 Mbit/s),  
full speeds (12 Mbit/s) and mixed

### Indicators

- Power: illuminated when the USB Tracker is powered on
- Activity: illuminated in green when packets are detected (in red when they are recorded)

### Power Requirements

- USB bus powered
- No external power supply is required

### Product Warranty

- Two years limited warranty

Ellisys reserves the right to change the features and specifications of the Tracker without notice.

## USB Tracker SDK

The USB Tracker 110 development kit makes it possible to easily write customised USB traffic analysis programs. The USB Tracker 110 sends USB packets and also bus and power states in real-time to the development kit for customized processing. Examples of usage include:

- real time display of data offered over USB,
- triggering of a process during a specific event on the bus,
- counting of error rates of a device under test,
- decoding of proprietary protocols,
- verification of the data integrity of a device under test,
- measurement of the response time of a device connected over a loaded bus.

The SDK is sold separately.

## Know more...

For more information about the USB Tracker 110, please visit the website dedicated to this tool:

[www.usbtracker.com](http://www.usbtracker.com)

## Overview

*The USB Tracker 110 is a USB protocol analyser allowing you to display the packets sent, decode the descriptors, detect errors in devices or drivers and measure their performances.*

*It is the ideal companion for anyone developing USB peripherals, embedded software or drivers. Its analysis and display software is easy to use so that you can quickly learn all about USB.*

*Connect the USB Tracker 110 between a personal computer or a laptop and any USB peripheral to instantly view the traffic. When capturing the packets, the real-time statistics display allows you to check the bus status even before you look at the packets which have been read. These are displayed in a chronological list together with the peripheral's address and the endpoint number. A second window gives details on the selected item.*

*To make it easier to identify the packets you are interested in, the software includes filtering and packet colouring functions. It also gives you a choice between several levels of detail and the information taken from the traffic comes with references to the USB standard.*

*The freeware display software allows you to display any previously recorded traces.*



## Advantages

- ✓ Low-speed (1.5 Mbps), full-speed (12 Mbps) and mixed capture
- ✓ Details on transactions and transfers in a second view
- ✓ Decoding and detailed display of standard requests and descriptors
- ✓ You can filter or colour packets with a click of the mouse
- ✓ Unlimited capture time using a high-speed USB 2.0 connection (480 Mbps)
- ✓ Real time statistics display during capture
- ✓ Didactic display with references to the standard
- ✓ Non-intrusive analysis
- ✓ USB bus-powered
- ✓ Small form factor
- ✓ Standalone freeware display application
- ✓ The decoding engine can be updated by software

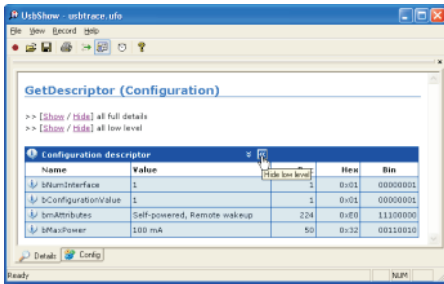
## Applications

The USB Tracker 110 is an ideal tool for supporting the development of embedded software, drivers or applications. It allows you to check all the stages in the listing of a USB peripheral by clearly displaying the various requests sent and showing any errors. The data capture and checking are both extremely user-friendly. Whether you want to analyse how an existing peripheral works, search for any incompatibilities or measure the performance of a driver or peripheral, the Tracker will give you full satisfaction.

## Alternative views

To allow the user to carry out his analysis without drowning in a sea of information, the application masks most of the data which is not necessary during normal use. In certain cases you may, however, need to display it.

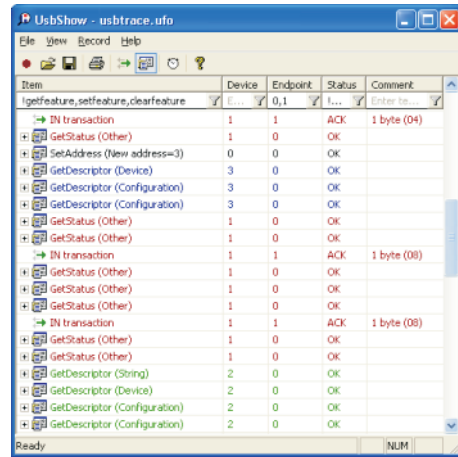
The descriptor data is formatted to make it easy to read, often in the form of texts or bit fields. If the user wishes to see a particular piece of information in decimal, in hexadecimal or in binary, he can do this with just a click of the mouse.



In the same way, certain fields, such as the byte length of a descriptor or character string indices, are masked by default but are always available by just clicking a button.

## Filtering

To get the most out of a protocol analyser, you need to be able to identify the relevant transactions. Although the amount of information displayed by the is kept down to the strict minimum, some applications do require the use of a filter. The USB Tracker 110 allows you to filter out unwanted packets with a click of the mouse. So you can mask or keep transactions of a certain type, certain peripheral addresses or certain endpoints. This allows you to eliminate SOFs (Start Of Frame) or NAK transactions.



If the filtered elements are definitely not going to be needed, you can save a copy of the file which does not contain these elements. When you next open this file, the previously filtered elements will not be shown.

## Colouring

When there are a lot of peripherals connected, the number of transactions displayed may make it fairly difficult to identify which one a particular transaction belongs to. To save the user from having to carry out tedious searches, the USB Tracker 110 allows you to colour transactions belonging to a given peripheral. So you can view the information sent to a particular peripheral in one colour, and see all the other communications in a different colour.

It can be particularly useful to use transaction colouring for educational purposes. For instance, you can compare the traffic from a hub to that of a peripheral which you are listing.

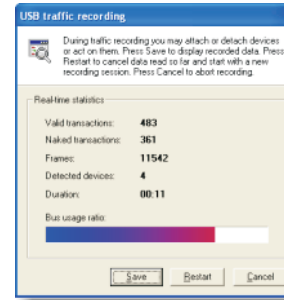
## Real time statistics

When recording the traffic on a peripheral you may need to find out how many of the transactions have been successful. The USB Tracker 110's capture dialog box displays and updates the following information in real time:

- The number of transactions with data
- The number of transactions without data
- The number of valid frames
- The total capture time
- The bus use rate

The user can record the data read, restart or cancel the capture at any time.

If he chooses to keep the packets read, they are grouped into transactions and displayed in the application's main window in a clear and comprehensible format.



## Grouping into transactions

The basic building block of USB is the packet but although necessary, it is often not the most interesting element. It is the transaction itself, generally made up of a number of packets, which actually carries the information. It represents an effective communication between a peripheral and its host. The Tracker's software displays a list of transactions with information shown in a number of columns so that you get an overview as soon as the capture is finished. In addition to the transaction type, you will find the address of the peripheral in question, the end-

« Learn all about your USB devices »

point number, the transaction status and in some cases a comment.

To make it even easier to use, the application analyses and decodes standard requests and descriptors and displays them in a simplified form. Using the USB Tracker 110, the user can view all the characteristics of the peripheral that he is developing in just a few seconds.

## Decoding engine

The decoding of the USB packets is carried out in a powerful programmable circuit. It can be reprogrammed for subsequent improvements. It is capable of automatically detecting the transmission speed (1.5 Mbit/s, 12 Mbit/s or mixed) during packet capture. The synchronisation byte and the EOP (End Of Packet) are used as triggers to start and stop the reading of the bits respectively. These are grouped into bytes and sent to the analysing computer via USB 2.0 (This transmission is carried out at high speed, 480 Mbit/s for best performances but full speed- can also be used when no USB 2.0 host controller is available).

Then the application analyses, decodes and formats the packets so that they can be clearly and comprehensibly displayed.

