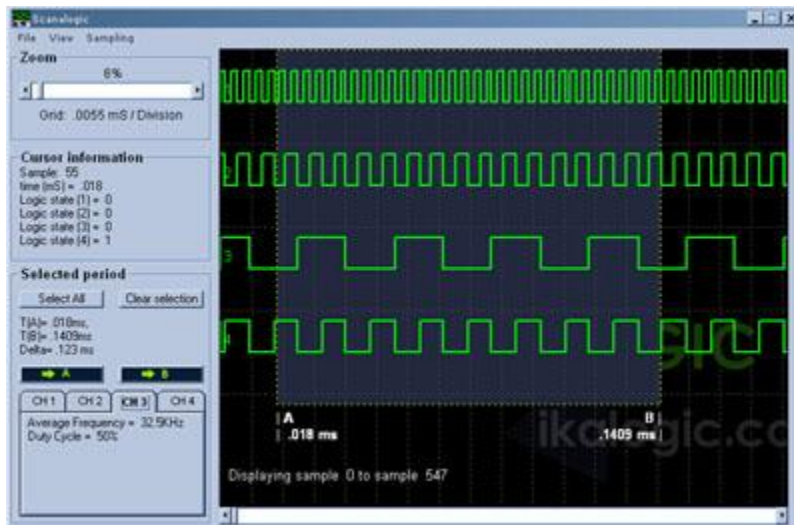


# Scanalogic: the beginning!

Posted on August 8, 2008, by Ibrahim KAMAL, in [Sensor & Measurement](#), tagged



*Here is one project we are really proud of, and we hope that its simple design will encourage you to build your own, and use the provided software to obtain your own logic analyzer for less than 10\$. This device can help you in most of your projects, it's an essential measuring device when it comes to digital electronics. Those pages will discuss all what you need to know to understand how it works, and build your own, and use it.*

## What is a Logic Analyzer?

You could read the definition on [Wikipedia](#), but as you may already know, I have my way of explaining things to beginner!

I would say that a logic analyzer is a device used to view the voltage of "logic" electric signals that are usually too fast to be measured by a voltmeter. A logic analyzer it not exactly like an oscilloscope, as it cannot measure analog voltages. Actually, it can only measure the logic state of a signal, see if it's a 1 or a 0. '0' being Zero volts and '1' being 5V (for classic TTL logic devices).

The main keypoint that makes a logic analyzer better than another is the sampling rate, or the maximum number of samples that can be recorded by second.

From a functional point of view, a logic analyzer records the states of logic signals, usually after some 'triggering condition' is met, then, recorded signals are displayed on a screen for in depth analysis.

## Introducing Scanalogic

Scanalogic is a simple, yet effective logic analyzer specially designed to be easily built by beginners, so that it can be used for debugging and analyzing their projects.

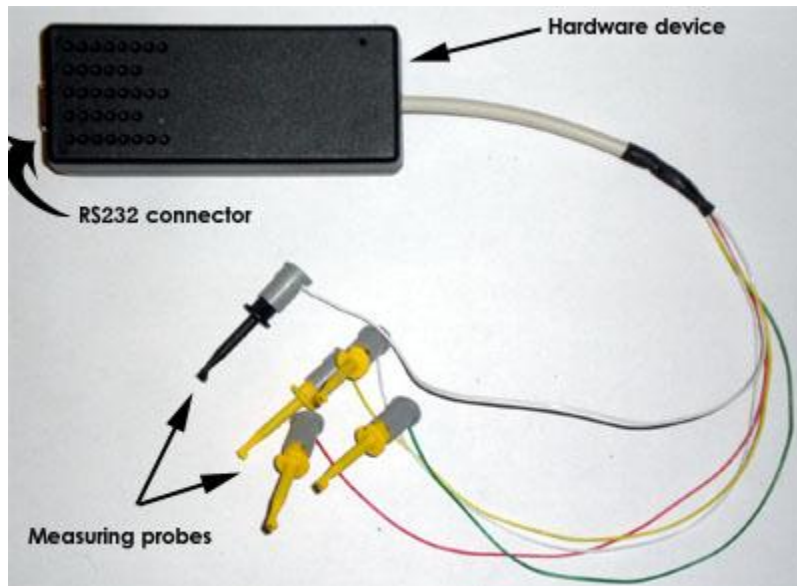


figure 1.A

It can be used at sampling rates as high as **4 Million Samples per second**. It has only four channels, but that is more than enough for most hobby projects, further more, most communication buses nowadays have very few transmission lines, ranging from two to four, so decided four was more than enough.

Scanologic is a combination of a hardware (**figure 1.a**) and software application (**figure 1.c**). As you will see later, the hardware is as simple as it can get: An ATMEGA16 micro controller and a NAND gate! The code to be loaded in the micro controller is provided at the end of this page.

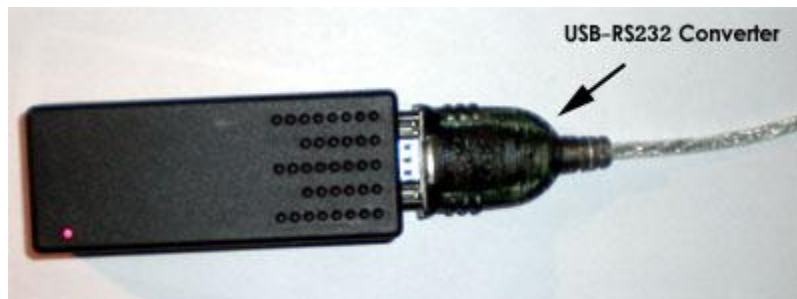


figure 1.B

The hardware records the samples at the desired sampling rate, then send them to the PC application via the UART port of the microcontroller and the RS232 port of the PC. If your PC doesn't have an RS232 port, as mine, you can still use a USB-RS232 converter cable, as shown in **figure 1.b**.

The software is made with visual basic. [A zip file is included at the end of the page.](#) It is compatible with windows ME all the way up to XP (or maybe Vista, can someone confirm?).

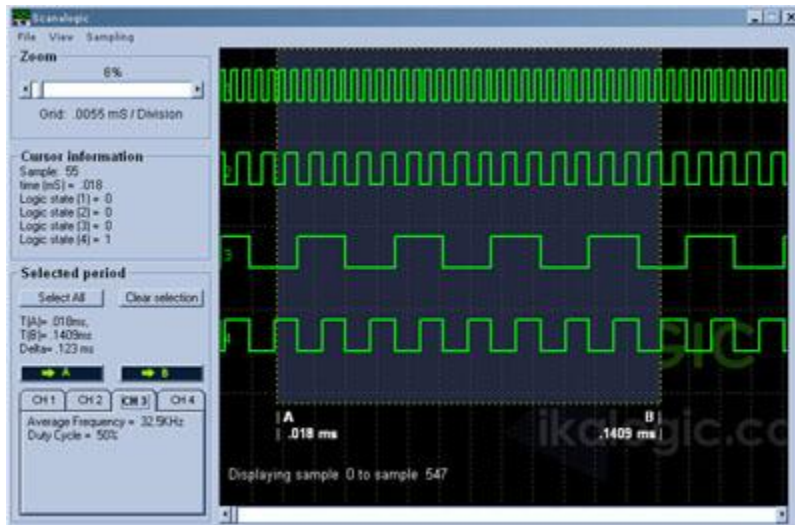


figure 1.C

The Scanlogic software controls the hardware (as you can see, there are no buttons on the hardware) It allows you to setup the triggering conditions, to start sampling, to receive sampled signals, display them, analyze them, and even store them for later analysis.

### Building the hardware

There are no secrets held for the hardware, as the real power of this logic analyzer comes from the innovating source code, loaded inside the ATMEGA16 microcontroller.

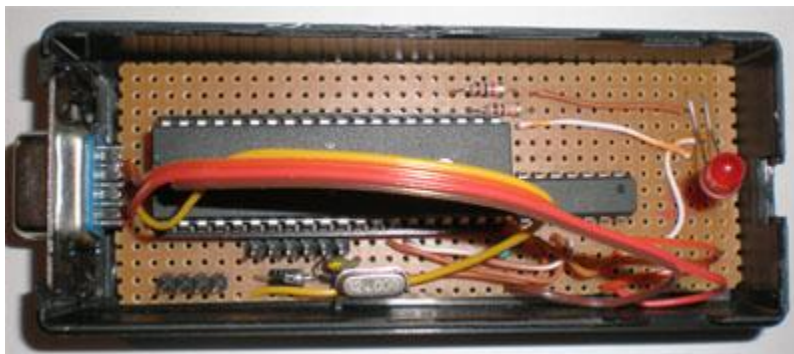


figure 2.A

As you can see in **figure 2.b**, there are only a microcontroller, and a NAND gates IC and all of it resides in an old CDROM power supply box. The DB9 connector for serial communication is firmly glued to the box with an epoxy resin. The whole is powered from the RTS line of the RS232 connection (pin number 4), so there is no need for external power supply (but you may as well use any 5V power supply if your serial port cannot deliver the required current).

As you can see in the schematic below (**figure 2.b**), there are no level shifting IC like the MAX232, but believe or not, most if not 100% of current serial communication hardware is compatible with the

approach below, where the signals are simply 'inverted', but not shifted to +/- 12V. (Originally, in the RS232 protocol, a logic 1 is represented with -12v, and a logic 0 is represented by +12V).

Pin 4 of the DB9 connector is used to power the device, as it rises to +12V (theoretically) when a communication is being carried out. The Diode D<sub>3</sub> makes sure no negative power supply is applied to the device, and the 5.1V zener diode D<sub>4</sub> prevent supply voltage from rising above 5.1 volts. Using the handshaking signals as a power supply is far from being my invention, it have been used in a lot of devices like mice and old external keypads...

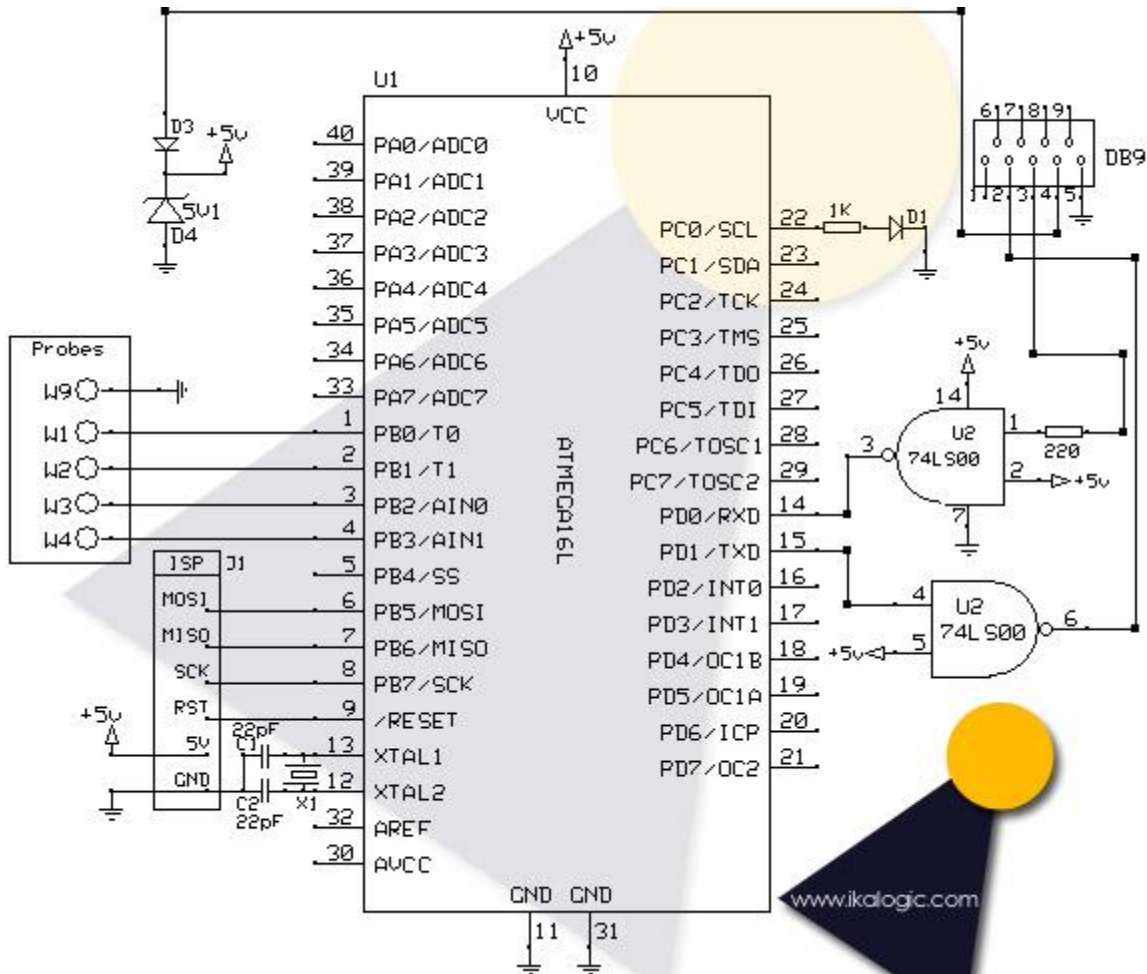


figure 2.B

The probes are directly connected to micro controller. A better approach would be to add buffers to protect the micro controller but, for simplification, I didn't include one.

The crystal oscillator X<sub>1</sub> can be anything from 1Mhz to 16Mhz, but remember that the clock frequency of the microcontroller directly affects the maximum sampling rate, according to the following equation:

$$\text{Max\_Sampling\_rate} = \text{Clock\_frequency}/4$$

That is simply because, the most efficient code I could ever write, needs at least four clock cycles to sample and store new signals.

D<sub>1</sub> is simply a red LED, used to signal if the device is powered up and correctly functioning.

***Download the hex file to be loaded into the ATMEGA16 for different crystal frequencies:***

- 8 MHz
- 12 MHz
- 16 MHz

*(Need hex files for other frequencies? request for it in the form in bottom page)*

### **Downloading and using the PC side software**

Scanalogic software, especially this first version is very simple and intuitive.

After you have plugged the Hardware, the configuration is simply made via 3 steps:

1. Press F2 to chose the crystal frequency at which your microcontroller is running, and to chose the COM port to which the hardware is connected.
2. Press F3 to chose the sampling rate.
3. Press F4 to chose the triggering condition.

Then you simply need to Press F5 anytime to start the sampling. Note that if you chose a triggering condition, the sampling will only start when this condition is met.

**Source: <http://www.ikalogic.com/scanalogic-the-beginning/>**