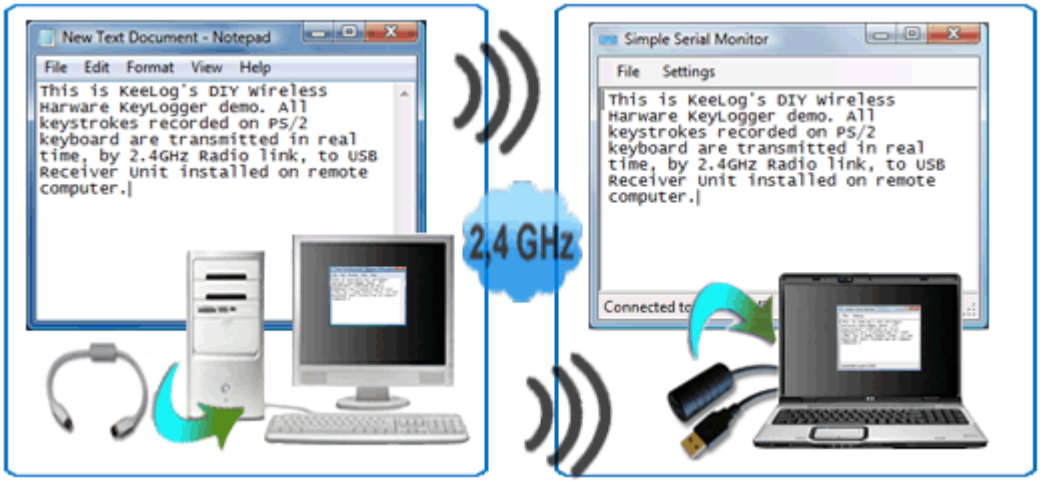
Wireless Keylogger

Do It Yourself!



Introduction

Familiar with the concept of hardware keylogging? A hardware keylogger is a perfect solution for monitoring user activity, at very low risk of disclosure. A hardware keylogger is a purely electronic device, so no access to the operating system is required, no traces are left, and software has no possibility of detecting such a device. However, the hardware keylogger concept inherits one weakness: physical access to the keylogger is required for retrieving captured data. This problem has finally found its solution: a Wireless Keylogger.

KeeLog has already released one open source PS/2 hardware keylogger design to the public. Now, we are doing it again with the DIY Wireless Keylogger. This design is fully free for private and commercial use, with the following restrictions:

1. All materials presented on this web page are the intellectual property of KeeLog and using them constitutes acceptance of the license terms below and the general User Agreement.
2. This Wireless Keylogger project is provided as is, with all faults, and with no warranty whatsoever.

You should not use the Wireless Keylogger to intercept data you are not authorized to possess, especially passwords, banking data, confidential correspondence etc. Most countries recognize this as a crime.

The Wireless Keylogger consists of two main building blocks: the transmitter, and the receiver. The actual keylogging takes place in the transmitter, which is in fact a PS/2 hardware keylogger, with a built-in 2.4 GHz wireless module. Captured keystroke data is transmitted through the radio-link in real-time, rather than getting stored. The receiver on the other hand, is a wireless acquisition unit with a USB interface. All keystroke data received from the transmitter is sent to the host computer via USB. From the software side, this data is available through a virtual COM port, allowing any terminal client to be used for visualizing keystroke data.

Wireless Keylogger block scheme

The entire system works in real-time, so text typed on the remote computer is seen immediately on the receiver side. The system has a maximum range of around 50 yards (meters). This corresponds to an effective range of around 20 yards (meters) through 2-4 walls, depending on their thickness.

Wireless Keylogger transmitter

Wireless Keylogger receiver

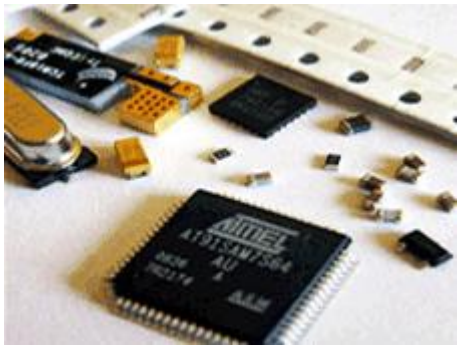
Both the transmitter and the receiver are based on the same schematics and circuit board. Both have the same form factor, and are intended for mounting on PS/2 and USB extension cables. The recommended housing is an EMC-balun enclosure, which makes the device resemble a standard extension cable.

Components

This article describes the entire assembly process of the DIY Wireless Keylogger. Depending on your skills, you may choose to create your own Wireless Keylogger from scratch, or order a preassembled one from us. We can deliver a set of components with pre-programmed microcontrollers and standard casing (as seen on pictures), or a fully assembled and tested set of devices. Please scroll to the kits section for more details.

If you decide to create your own Wireless Keylogger, you should have some basic experience with electronics and soldering, ideally with SMT (Surface Mounted Technology). The easiest option includes ordering a kit with components from us, and doing the soldering, cabling, and final assembly on your own. This involves having a temperature-controlled soldering iron and quite good soldering skills. If you decide to design and produce the PCBs yourself, you should have significant experience in this field and proper equipment.

The table below summarizes the BOM (Bill of Materials) contained in a single transmitter or receiver unit. An additional PS/2 extension cable is required for the transmitter, and a USB type A connector or cable is required for the receiver.



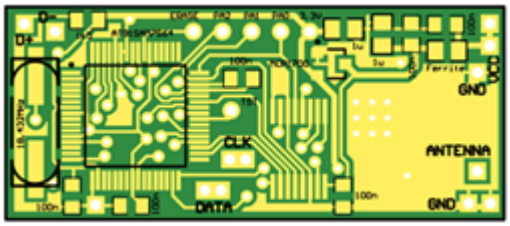
Set of electronic components

Cables, enclosure, and PCBs

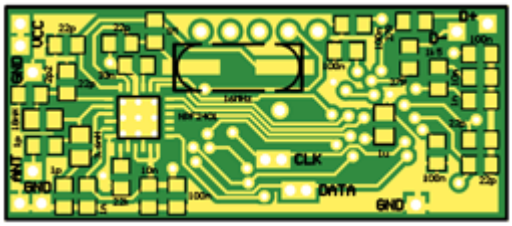
Designator	Description	Footprint	Qty
U1	Microcontroller AT91SAM7S64	TQFP64	1
U2	Transceiver nRF2401	QFN24	1
U3	Voltage regulator MCP1700T-330	SOT-23	1
Q1	Crystal 18.432 MHz	HC-49 SMD	1
Q2	Crystal 16 MHz	HC-49 SMD	1
R1, R2	Resistor 1.5 kΩ	0805	2
R3, R4	Resistor 27 Ω	0805	2
R5	Resistor 1 MΩ	0805	1
R6	Resistor 22 kΩ	0805	1
C1, C27	Capacitor 10 nF	0805	2
C2, C28	Capacitor 1 nF	0805	2
C3, C4, C6, C7, C8	Capacitor 22 pF	0805	5
C5	Capacitor 33 nF	0805	1
C9	Capacitor 2.2 pF	0805	1
C10, C11	Capacitor 1 pF	0805	2
C12, C22, C23, C24, C25, C26, C32, C33, C34, C42, C43	Capacitor 100 nF	0805	11
C21, C31, C41	Capacitor 1 µF	0805	3
L1	Ferrite Bead	0805	1
L2	Inductor 3.6 nH	0805	1
L3	Inductor 18 nH	0805	1

Wireless Keylogger BOM

Both the transmitter and the receiver use the same PCB and the same set of components (they differ by cabling and firmware). The Atmel AT91SAM7S64 microcontroller and the nRF2401 wireless transceiver are the core components. Both require crystals for proper operation. Besides the MCP1700 voltage regulator, all other components are passive (resistors, capacitors, and a few inductors). A simple wire is recommended for the dipole antenna. The double-sided two-layer PCB is shown on the pictures below.

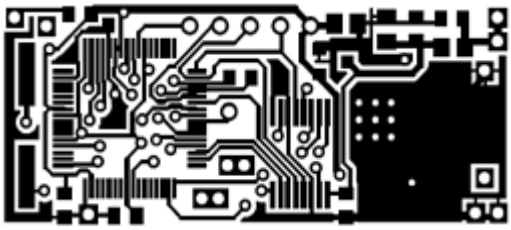


PCB layout - top side

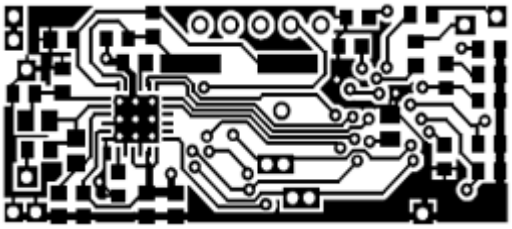


PCB layout - bottom side

If you feel skilled enough to manufacture PCBs yourself, you may use the 1:1 mask set available below. The reference design uses FR4 with 1.0 mm thickness.



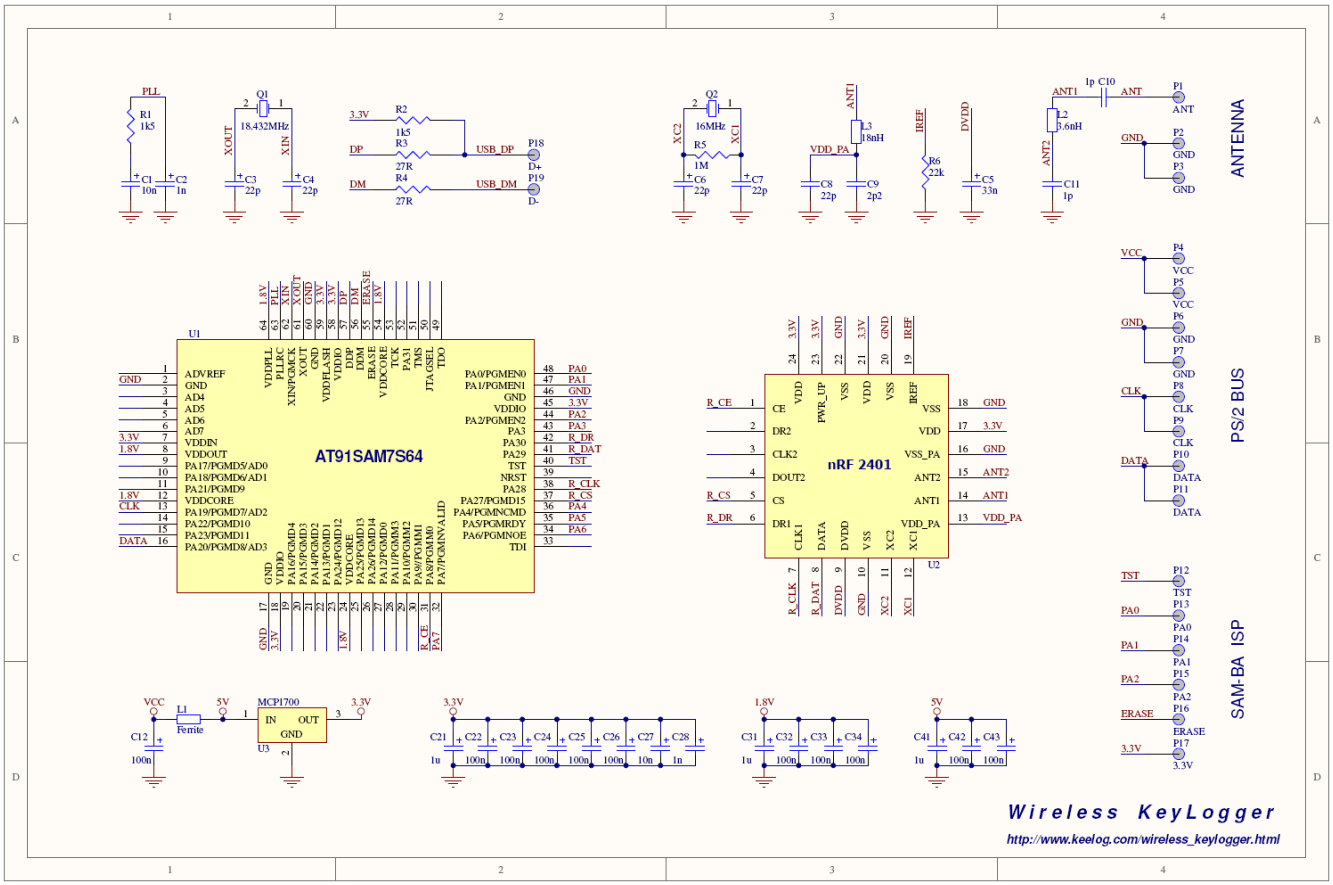
PCB mask - top side



PCB mask - bottom side

Assembly

The Wireless Keylogger electrical circuit is composed of two main building-blocks: the AT91SAM7S64 microcontroller and the nRF2401 transceiver. The accompanying passive components are mainly oscillator and RF circuitry. The entire circuit is powered with 3.3V, generated by the MCP1700 regulator and filtered by a set of capacitors. Power is drawn directly through the PS/2 bus (transmitter), or USB (receiver). If you already have assembled mini-boards, skip to the wiring section. If you have decided to assemble the circuit boards yourself, you'll need to follow the schematics and guidelines below.



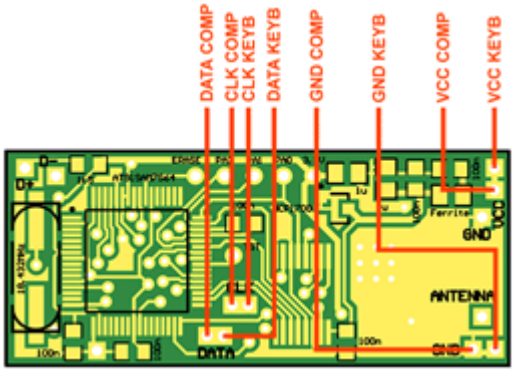
Wireless Keylogger electrical schematics

Use a fine tip for soldering (typically smaller than 0.5 mm) and soldering flux (for example RMA7). Don't overheat the components. Start the assembly with the nRF2401 transceiver, as it has the most difficult footprint. Proceed with the AT91SAM7S64 microcontroller and the MCP1700 voltage regulator. Always make sure pin 1 matches the first pad on the PCB. Finally, solder all the auxiliary circuitry: crystals, resistors, capacitors, and inductors. Leave the antenna for the end. You can use a dedicated ISM 2.4 GHz antenna, or simply make a quarter-wave dipole antenna from a piece of wire. The optimal length is 1.23" (3.125 cm). Assembled mini-boards should look similar to the ones on the pictures below.

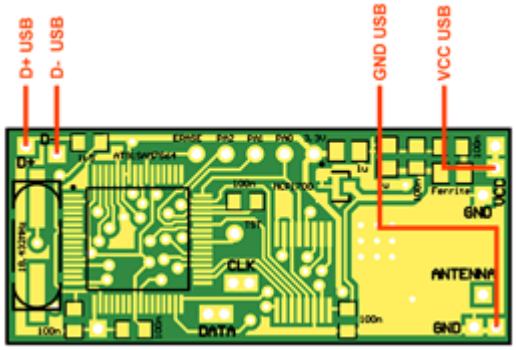
Assembled PCB top side with microcontroller

Assembled PCB bottom side with transceiver

After assembling the circuit boards, it's time for the cabling. Apart from firmware, this is the place where the transmitter differs from the receiver. The transmitter should be coupled in parallel with the PS/2 bus. The PCB has pads for connections leading both to the computer, and to the keyboard. The receiver, on the other hand, should have a standard connection to the USB port. The images below show how the connections should be made.



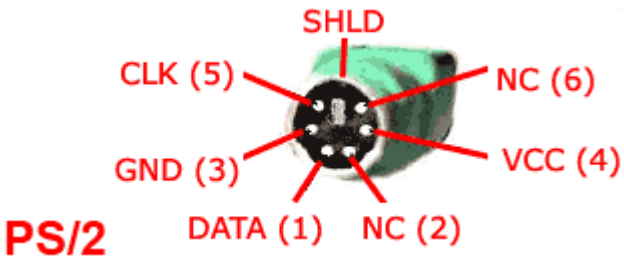
Transmitter PS/2 wiring diagram



Receiver USB wiring diagram

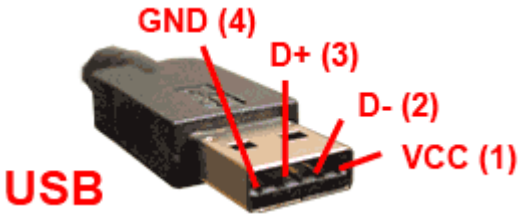
Use PS/2 and USB extension cables, cut them open, and isolate the signal lines. The tricky part is to identify how the wires inside correspond to signal lines. Some PS/2 and USB cables have standardized colors, however trusting in this is very risky. The recommended solution would be to use a short-circuit tester or ohmmeter to find out which wire corresponds to which signal. The diagrams below will be helpful.

Signal	Description	PS/2 pin	Comments
VCC	+5V power	4	must be connected to module
GND	Power ground	3	
CLK	Clock	5	
DATA	Data	1	
NC	Unused lines	2, 6	not used by module if present, leave in original state
SHLD	Shield	-	



PS/2 plug pinout (transmitter unit)

Signal	Description	USB pin	Comments
VCC	+5V power	1	must be connected to module
D-	Data	2	
D+	Data	3	
GND	Power ground	4	
SHLD	Shield	-	not used by module if present, leave in original state



USB plug pinout (receiver unit)

If the microcontrollers you're using aren't programmed yet, this is a good moment to upload firmware using the ISP (In-System Programming) technology. Read the firmware section to get more details. When this is done, the mini-boards should look similar to the ones on the pictures below.

Transmitter circuit board wired to PS/2 bus

Receiver circuit board wired to USB

Before putting the enclosure on, we recommend to do one last check. Use a short-circuit tester or ohmmeter to check the resistance between the power supply (VCC) and ground (GND) on the USB and PS/2 connector. The presence of a short circuit here means, that the whole circuit should be revised, otherwise it could lead to damaging your computer. If everything's OK, mount the enclosure using glue or resin, and you're set to go.

Power-up

Once you have a transmitter-receiver pair of devices assembled, it's time for the first test. We recommend to use a single computer for testing both devices. First, power down the computer and connect the transmitter unit between the PS/2 keyboard and PS/2 port.

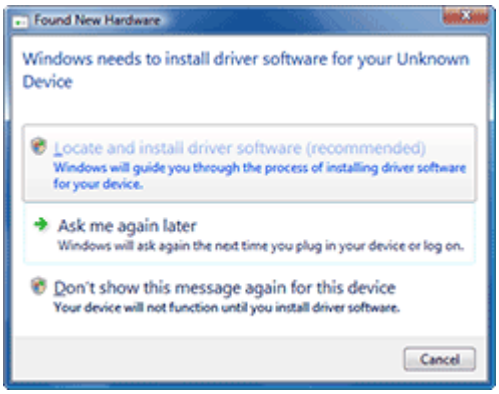
Connect the transmitter unit to the PS/2 port

Connect the PS/2 keyboard to the transmitter unit

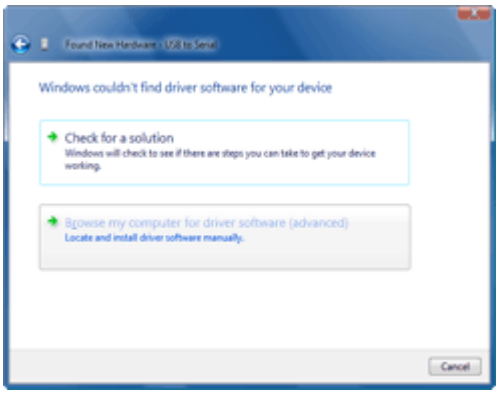
When done, boot the computer and make sure the PS/2 keyboard is working properly (no influence of the keylogger should be noticeable). Now it's time to test the receiver unit. Before you proceed, please download the KeeLog driver files first. Unzip and save the files to the local hard disk on your computer. Then, plug the receiver unit into a free USB port (no need to power down the computer). Make sure it's in a position enabling reception of the radio signals coming from the transmitter.

Connect the receiver unit to a free USB port

The first time the receiver unit is connected, a driver installation dialog will appear. Strictly speaking, it will use the bundled virtual COM port driver delivered with most operating systems, such as Windows. However, the corresponding INF description file has to be selected manually. When the system asks for a driver, browse to the location where the driver file was saved. The pictures below illustrate the process.

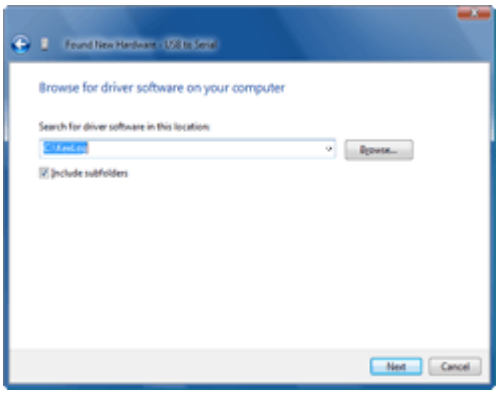


Choose to locate and install driver software



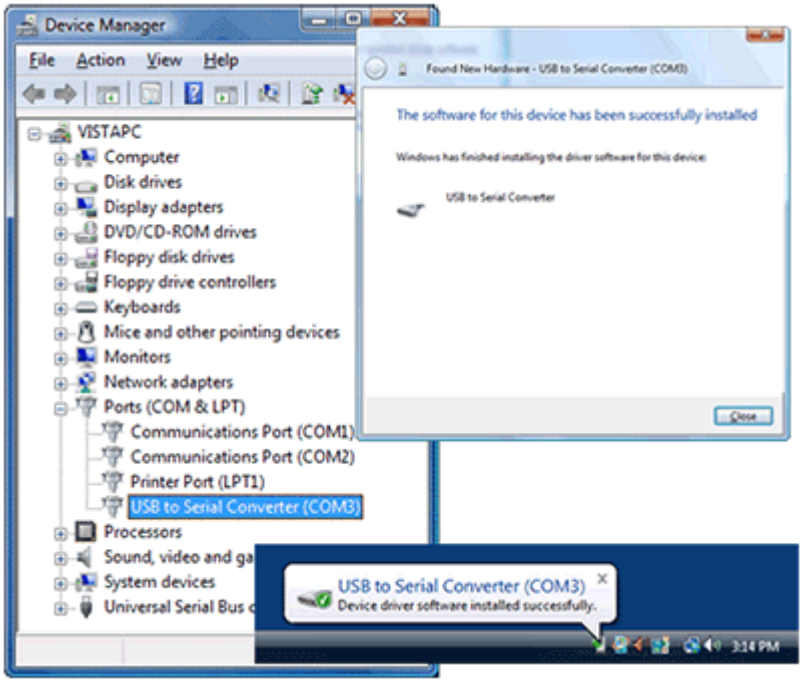
Choose to browse for driver

Choose to show browse option



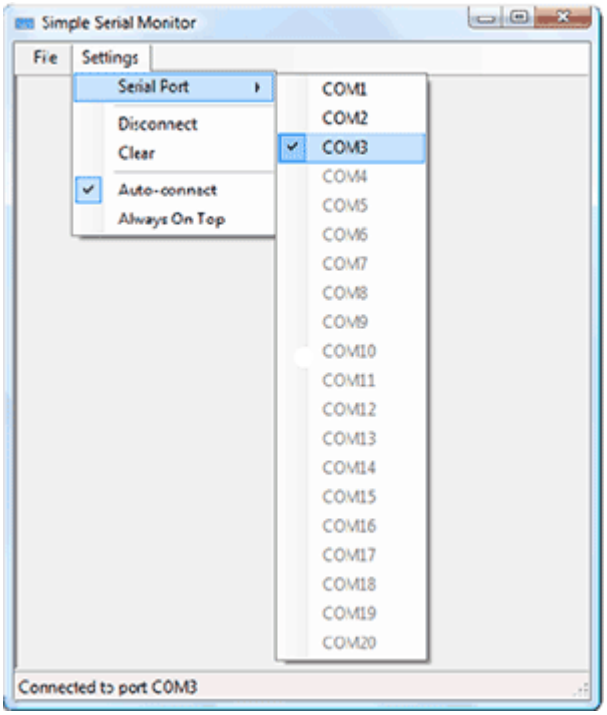
Browse to driver location

If the driver installation process was successful, the receiver unit should be visible as a USB to serial converter. Open the Device Manager in Windows to find out, and check which virtual port was assigned to the device.



Receiver unit visible in the Device Manager

To start receiving keystroke data from the transmitter unit, you may use any terminal client, such as Hyperterminal. We recommend to use our free Simple Serial Monitor application for it's flexibility and ease of use.



Simple Serial Monitor (free terminal client from KeeLog)

After launching Simple Serial Monitor (or any alternative application), remember to set the correct COM port. If everything proceeded correctly, the receiver unit will immediately show all keystrokes typed on the PS/2 keyboard.

Remote computer with PS/2 transmitter unit *Local computer with USB receiver unit*

The next step would be to test the same using two different computers. Make sure they are in transmission range. If you see text popping up in the terminal window, then your Wireless Keylogger set is ready for its first real mission. Remember to use this device only for legitimate purposes!

Download

Microcontroller firmware for programming the transmitter and receiver
http://www.keelog.com/files/WirelessKeyloggerFirmware.zip

Driver allowing the receiver to be recognized as a virtual COM port
http://www.keelog.com/files/UsbToSerial.zip

Free software for displaying intercepted keystroke data through the virtual COM port (equivalent to Hyperterminal). Requires the Microsoft .NET Framework.
http://www.keelog.com/files/SimpleSerialMonitor.zip

Software for flashing firmware using the SAM-BA bootloader
http://www.keelog.com/files/At91Isp.zip

Tutorial on flashing the firmware into the microcontroller through a built-in bootloader, without using any additional programmer
http://www.keelog.com/files/SambaUserGuide.pdf

List of components for the assembly of the Wireless Keylogger (transmitter and receiver)
http://www.keelog.com/files/WirelessKeyloggerBom.pdf

Wiring scheme for the Wireless Keylogger (transmitter and receiver)
http://www.keelog.com/files/WirelessKeyloggerWiring.pdf

Electrical schematic for the Wireless Keylogger (transmitter and receiver)
http://www.keelog.com/files/WirelessKeyloggerSchColor.pdf

Top side of the PCB layout (transmitter and receiver)
http://www.keelog.com/files/WirelessKeyloggerPcbTop.pdf

Bottom side of the PCB layout (transmitter and receiver)
http://www.keelog.com/files/WirelessKeyloggerPcbBottom.pdf

Mask for the PCB top side (transmitter and receiver), scaled 1:1
http://www.keelog.com/files/WirelessKeyloggerMaskTop.pdf

Mask for the PCB bottom side (transmitter and receiver), scaled 1:1
http://www.keelog.com/files/WirelessKeyloggerMaskBottom.pdf

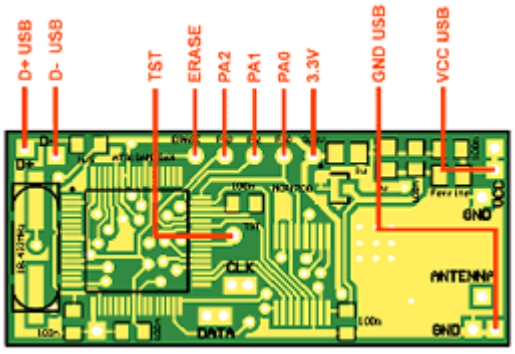
Firmware

Read this section only if you need to flash the AT91SAM7S64 microcontroller by yourself. If you have purchased a kit from us, we have already done this for you.

Modern microcontrollers, such as the Atmel AT91SAM7S64 have highly-packed footprints, making it difficult to find traditional programmers supporting them. That's why ISP (In-System Programming) has developed very rapidly in the recent years. ISP allows for assemble the entire circuit board first, and then flash the firmware, often using very simple tools. The AT91SAM7S64 implements a very convenient ISP solution, based on the built-in USB module. It's called the SAM-BA (SAM Boot Assistant), and requires only a USB cable and a few simple jumpers. To run SAM-BA on your Wireless Keylogger mini-boards, first download the AT91 ISP tool. Then, follow the steps below to complete firmware flashing on the transmitter and receiver unit.

Step 1: Applies for the transmitter unit only. Prepare a USB cable with a type A male plug on one side, and isolated wires on the other side. Solder the USB lines VCC, GND, D+, and D- to the appropriate pads on the PCB. This step is not necessary for the receiver, as it already has a USB connection.

Step 2: Prepare a few short wires for short-circuiting the SAM-BA pins: TST, ERASE, PA2, PA1, PA0, 3.3V. Solder one end of each wire to the SAM-BA pads on both boards. Alternatively, you may prepare special jumpers as seen on the pictures.



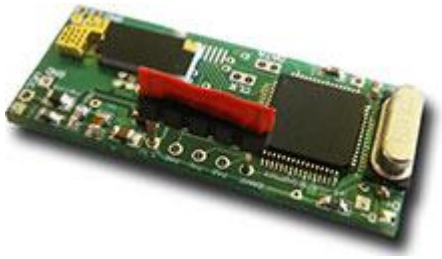
SAM-BA wiring scheme

Step 3: Install the AT91 ISP software package.

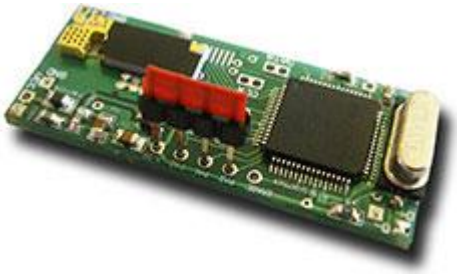
Step 4: Connect the device to a free USB port. A Device Not Recognized message is normal at this stage.

Step 5: Short the ERASE and 3.3V signal wires for a moment. This will erase the microcontroller's flash memory.

USB cable and jumpers for SAM-BA bootloading



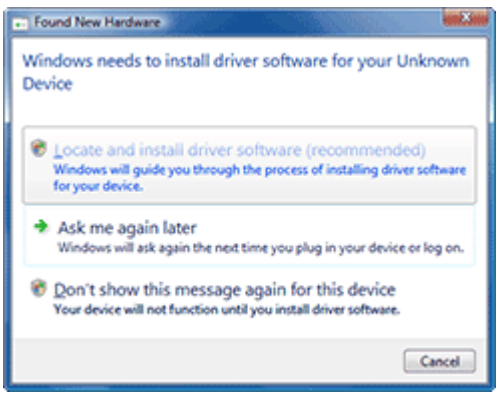
Memory erasing (ERASE pin shorted to 3.3V)



Bootloader activation (PA0, PA1, PA2 and TST shorted to 3.3V)

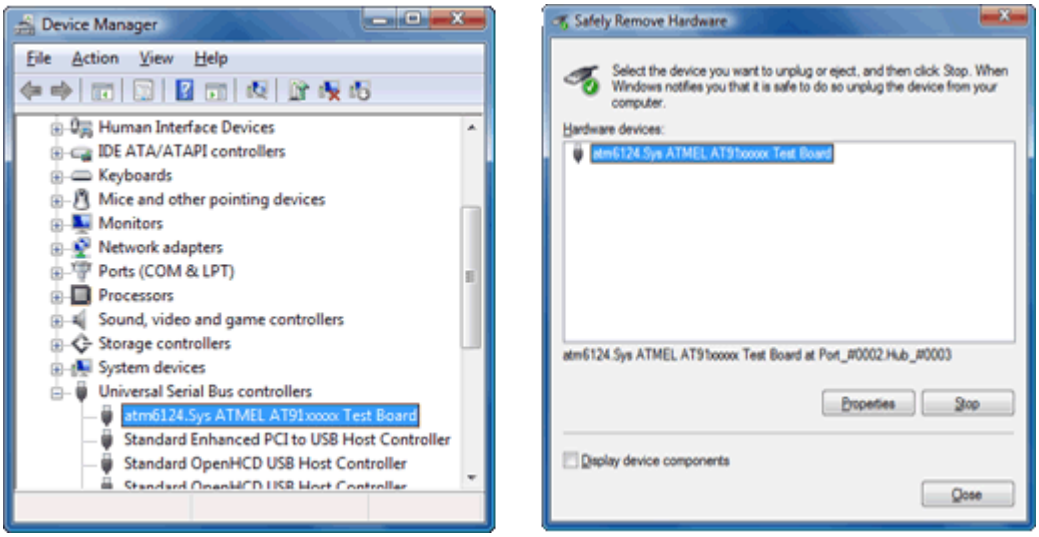
Step 6: Disconnect the device from the USB port. Make sure the ERASE pin is not connected to 3.3V any more. Now short the set of pins PA0, PA1, PA2 and TST to 3.3V. Connect the device to the USB port again (Device Not Recognized may appear again). Leave the device connected for approximately 10 seconds, and then disconnect the device from the USB port. This operation should have activated the SAM-BA bootloader.

Step 7: Remove all shorts or jumpers and connect the device to the USB port. The New Hardware Found dialog should appear. Please follow the default procedure and allow the wizard to find the drivers itself.



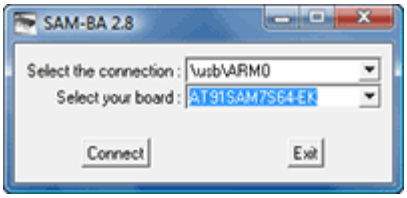
Found New Hardware wizard

Step 8: Open the Device Manager and verify that the SAM-BA bootloader has been activated.



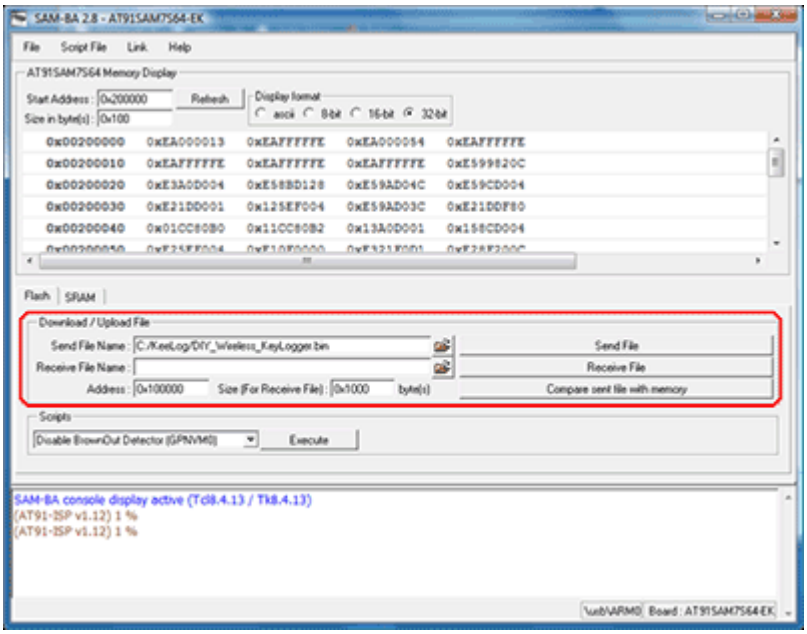
Device Manager with Atmel AT91 device

Step 9: Run the SAM-BA application from the AT91 ISP software suite and select the AT91SAM7S64-EK target microcontroller board.



Microcontroller board selection

Step 10: After establishing the connection with the board, switch to the Flash tab, select the appropriate firmware for the transmitter/receiver, and click on Send File. When the application asks whether to lock and unlock the involved flash regions, select yes. If you were successful in finalizing this step, it means the firmware has been downloaded to the microcontroller.



SAM-BA Main Window

Remember to go through the SAM-BA procedure for both the transmitter, and the receiver. When finished, both devices are ready to go.