

How to revive a Cellvision 670/WL-5460CAM/compatible WebCam after failed firmware upgrade

by Simon Hradecky, September 15th 2007
updated July 9th 2012 with updated link

It is highly risky to flash new firmware onto the WebCams Cellvision 670/670W and identical OEM webcams like the Sparklan CAS-370/370W, Air Live WL-5460CAM, Trendnet TV-IP400, Digitus DN-16030/ DN-16031 and many others. Quite frequently it happens, that the camera reboots in the middle of the firmware upgrade, leaving the firmware in a status, where the system can no longer boot, can no longer be reached on the network and seemingly can no longer be reflashed to get back to function.

The camera is driven by an ADM5120 microcontroller running a system called „BusyBox“ on top of Linux, which features a serial port driven by a separate boot loader, that is not being affected by firmware upgrades. So there's the chance to get a dead camera back to life, if a firmware flash went foul. The good thing about: that serial port is not only just useful for reflashing the firmware, but actually is a console to the Linux operating system, so that you can look into the running Linux and issue commands like ps, ls, netstat, ifconfig, ...

The following description is provided without any warranty attached. Continue at your own risk!

1. Important Notices

In principle you did forfeit your warranty already by flashing a new firmware onto your camera by the warranty regulations imposed by the manufacturer of your camera. But now you are going to open the camera, which definitely voids any warranty.

The firmware version to be uploaded through the webserver can not be used for upload via the bootloader on the serial port and vice versa – these are different files!

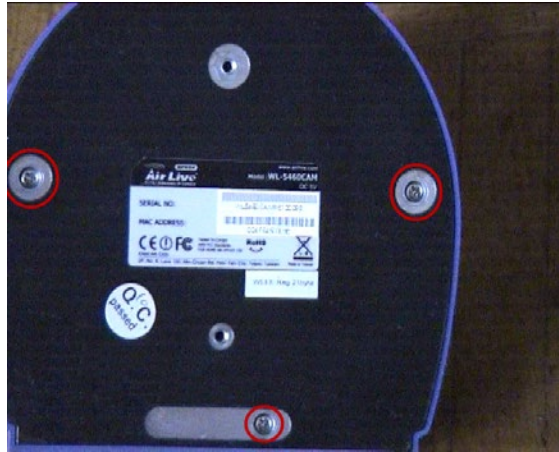
2. Prerequisites

You'll need to following prerequisites to get the camera reflashed:

- a) the file „vmlinuz“ (the Linux boot loader), that is available for example from AirLive at their published GPL licensed development kit to enhance/reprogram the cam at http://fs.airlive.com/GPL_Code/GPL_IP-CAM_5400_5420_5460.zip (103MB download)
- b) A TTL/CMOS to RS-232/V.24 signal converter (see below for a wiring diagram of a self made converter)
- c) A 4-pole connector (socket plus plug at 1.25mm/0.049“ raster), e.g. RS-Online's (e.g. <http://www.rsonline.at>, <http://www.rsonline.de>) order number 279-9162 (plug), 279-9437 (cables including contacts) and 279-9229 (head).
- d) RS232/V.24 cable
- e) PC running a terminal program capable of X-MODEM Protocol (e.g. Hyperterminal available with any Windows PC)

3. Preparation of the camera

You'll need to open the bottom of the camera first by removing the three screws, that are hidden underneath the rubber covers:

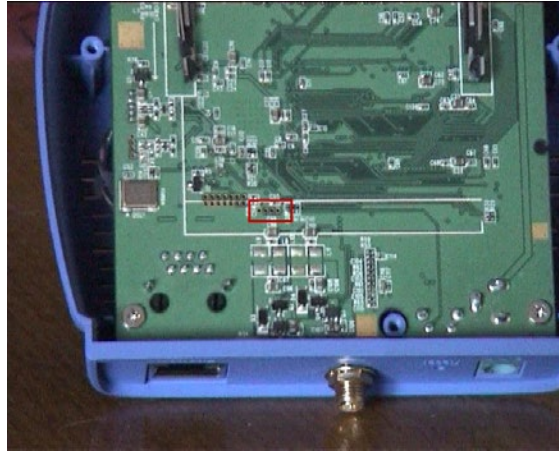


then turn the camera around and shake the floor plate out of the camera. You are now looking at the main board of the camera and locate the prepared connector for the serial port, that however has got no head soldered in:

with the WLAN module present:

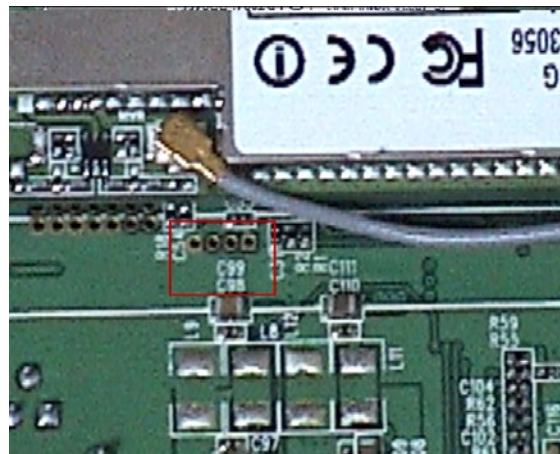


without the WLAN module:

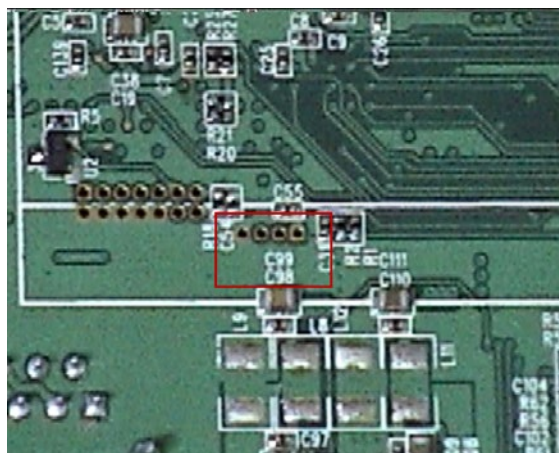


You find 4 contacts without any label above (as seen in the photos above) C99/C98, the most right contact being square (marking pin number 1):

With WLAN module:



Without WLAN module:



If you like, you can now solder the 4-pole head mentioned above into the board, best in a way, that the head points towards the bottom of the camera (so that you don't need to unscrew the board, if you want/need to use the serial port again).

The Pin Layout is (as seen in the photos above):

Pin 1 (most right pin): +5V

Pin 2: Tx (Cam transmits data on this line)

Pin 3: Rx (Cam receives data on this line)

Pin 4 (most left pin): ground

If you are going to use the 25 pin RS-232 connector, pin 2 of the CAM needs to connect to pin 3 of the RS-232 connector via the TTL/V.24 converter, and pin 2 of the RS-232 connector needs to connect to the pin 3 of the CAM also via the TTL/V.24 converter. The converter can be supplied with power from pins 1 and 4 by the camera.

Make sure, while working on your cam or connecting the serial cable to your cam, that the cam is not powered at all. Keep the cam unpowered until later.

4. Preparation of the PC

Extract the file vmlinuz from the downloaded GPL package. Run WinZip to open the downloaded file. You will find 4 tar.gz files in the package, the one you need is linux-2.4.18.tar.gz, which you simply open by double clicking on it. WinZip now extracts that file into a temporary folder and shows the contents of the file. You'll find the file vmlinuz now at a size of approximately 3.6MB. Extract this file and keep it ready for use within the terminal program.

Now connect the serial cable (that connects to your camera via the converter) to your PC (e.g. COM1) and launch the terminal program. Adjust baud rate to 115200 baud, 8 bits, no parity, terminal emulation ANSI or VT-100.

5. Reflashing the firmware

With the terminal program active on your PC, and your fingers close to the keyboard, now power your camera up (with the other hand). You'll get to see a message almost instantly:

ADM5120 Boot (V1.04)

As soon as this message appears, quickly press the space key three times (you have about 1 second or so to the first keypress). If you are too slow, you may see the following lines coming up:

LINUX/5120 started...

CPU revision is: 0001800b

Primary instruction cache 8kb, linesize 16 bytes (2 ways)

And so on. You may also see nothing – depending on which state your cam has arrived at – and experience your camera react to no input at all.

If you were quick enough, you'll see:

ADM Bootloader (v0.04.01 20040216)

=====

- (a) Download vmlinuz to flash ...*
- (b) Download vmlinuz to sdram (for debug) ...*
- (c) Update bootloader ...*
- (e) Exit*

Please enter your key :

Now press a to start the download of the vmlinuz file we extracted earlier. Immediately thereafter start sending the file using XMODEM - on hyperterminal click the “send” button, select “X-MODEM), enter the file path and name (perhaps you keep that path and name in your clipboard), and start sending the file. You’ll now get a screen showing you, how many packets have already been uploaded.

Once the file has been uploaded, the cam will tell you, that the flash memory is being erased, then it’s being written indicating, it has passed the flash with the message “PASS”. Then it returns to the menu above. You now exit by pressing e and get to see again something like:

LINUX/5120 started...
CPU revision is: 0001800b
Primary instruction cache 8kb, linesize 16 bytes (2 ways)

Now your camera is back to life and has a valid firmware, it will soon be seen on the LAN again. Note, that most likely all your settings have returned to factory setting, so the camera becomes visible at 192.168.1.2 (AirLive) with no WLAN active and your usernames and passwords reset as well. It also contains a firmware version, that is not commercially available (that firmware version is equivalent to Cellvision firmware 1.70). You may now want to flash your wanted firmware (e.g. Cellvision/Sparklan 1.71, or AirLive 1.01) onto the camera using the webserver.

Note: the firmware version to be uploaded through the webserver can not be used for upload via the bootloader on the serial port and vice versa – these are different files!

6. TTL/CMOS-V.24/RS232 Converter

If you don’t have a TTL/CMOS-V.24/RS232 Converter at hand, aren’t able to get a ready made product like available via <http://www.superdroidrobots.com/shop/category.asp?catid=42> (check Google with the search term ‘TTL RS-232 Converter’ for more products and sources), you can build your own one with little effort and money using the MAX232 chip. You’ll need one integrated circuit MAX232, 5 capacitors 1µF (depending on version of the MAX-232 chip those capacitors might be 10µF or 0.1µF), one experimental board and some soldering equipment ...

