Ultimate Interface for Amstrad CPC (ULIfAC)

User's Guide v3.0

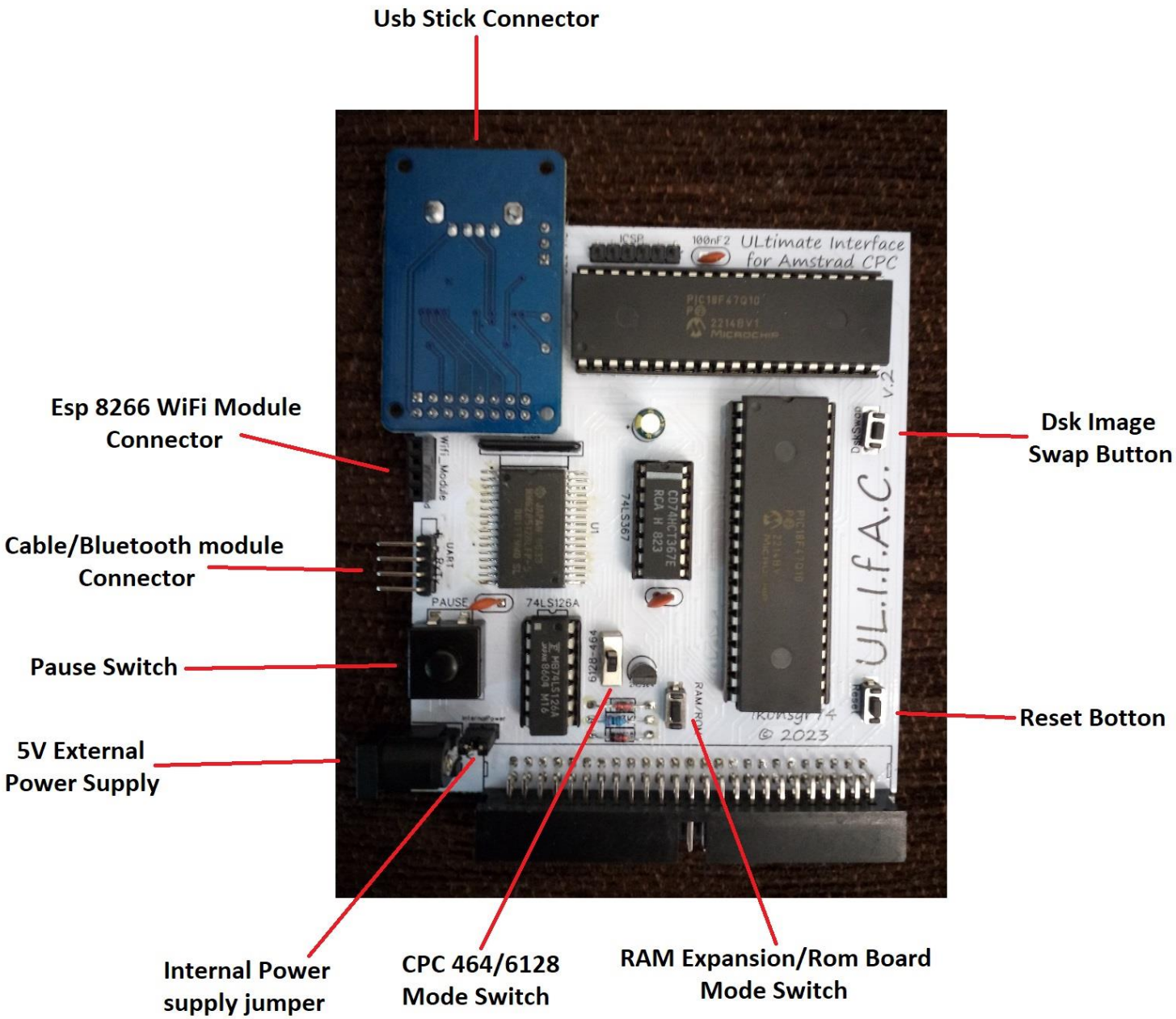
(John Konstantopoulos, August 2023 ikonsgr745@hotmail.com)

Table of Contents

Introduction

ULtimate **I**nterface **f**or **A**mstrad **C**PC (a.k.a. ULIfAC), is an evolution of **USIfAC II** board, now offering many new amazing features:

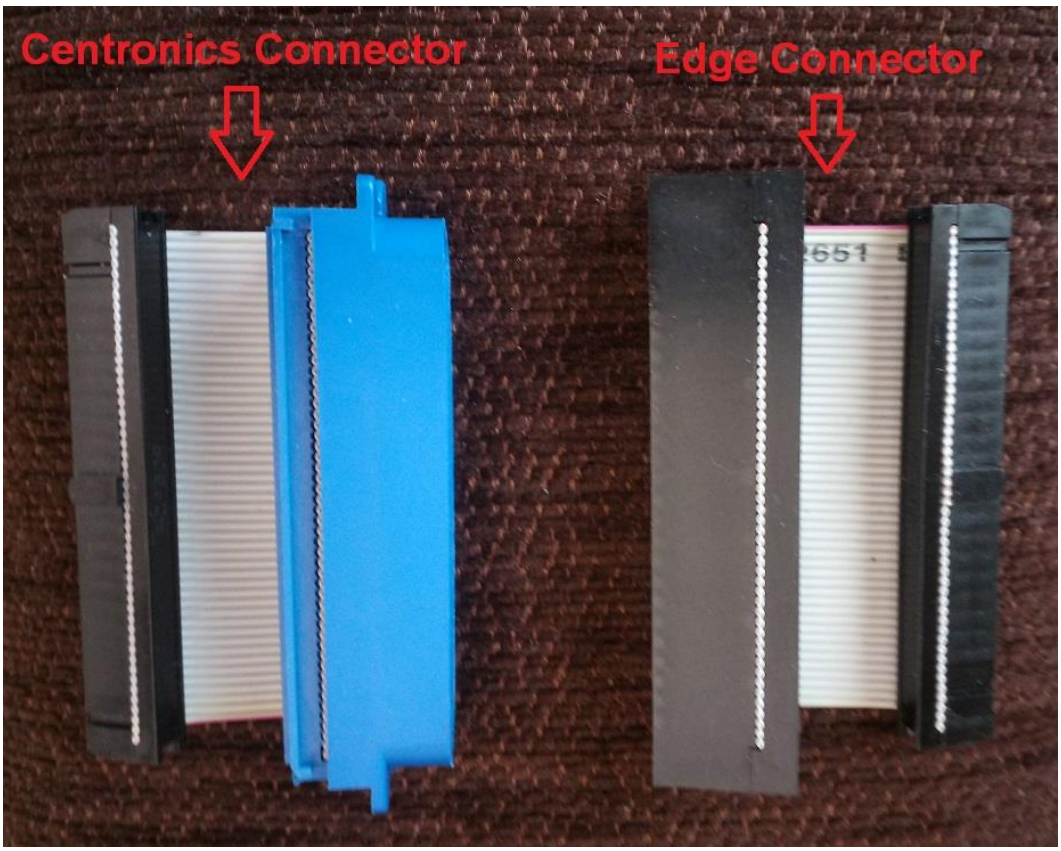
- **512Kb Ram Expansion**
- **32X Rom Board**
- **256Kb Ram Expansion + 16X ROM board**
- **Convert instantly a CPC 464 to a CPC 6128** using a simple RSX Command
- **Automatic Rom mounting upon cold boot!**
- **Many new RSX commands,** including instant loading of File manager, Diagnostic tool, Parados, text/ascii file viewer and other:



Of course all USIfAC II features are still provided:

- **Low level 765 Floppy Disk Controller emulation**,
- **Use a Usb stick as mass storage device**
- **Full access of PC's Hard Drive or usb storage drive**, for **change/create/delete directories and load/save/delete programs/games even Ascii files!**
- **Incredible loading speeds** of up to ~30kb/sec (**6-7 times faster than a real disk drive**)!
- **Full read/write access to DSK images** of both AMSODS and PARADOS formats!
- **Access up to four DSK images and choose them "on the fly",**to support all multi-image Games!
- **Load .SNA snapshot files!**
- **Copy files from/to floppy disks, transfer DSK images to floppy disks, backup floppy disks to dsk image files, and quick Format disks.**
- 1kb EEPROM for use as extra memory or for storing/executing your own routines.
- Includes 3 classic games**, GALACHIP, PACMAN and KILLER GORILLA**!
- Equipped with a **reset button and a Pause switch**

You can connect it directly to an external port raiser adapter like MX4 board, or directly to Amstrad's Expansion port using these ribbon cable adapters:



There are 3 Modes of operation:

- **512K RAM Expansion** (default mode)
- **32X Rom board mode** (supports both lower and upper roms)
- **Dual 256K Ram expansion + 16X Rom board mode** (supports both lower and upper roms)

Swap between Ram mode and Rom/Dual mode can be done by pressing the RAM/ROM button or by giving **|SW** RSX command. You can use |ROM rsx command to load Rom configuration utility for select ROM mode/ Dual mode, enable/disable Rom slots,activate lower rom etc. See **"Utilizing ROM Board/Dual Mode of Operation"** section for further details.

Now, as officially CPC 464 doesn't support RAM expansions, board uses a "trick", in order for the extra RAM to function: It suppresses MREQ signal, whenever a write to external RAM is occurred. The small "464-6128" switch on the board, enables/disables this "trick", so **if you use board on a CPC 464 you must set switch to '464' position.**

Although the above method seems to work fine in most cases (e.g. mostly games),some applications like SYMBOS and Future OS don't seem to "like" this trick, and will not load on a CPC 464. Also C3 mode is currently not supported, so CPM PLUS can't be loaded on CPC464 either.

Quick Start Guide:

You can directly use **File manager** by giving RSX Command:**|FM.** (note: use **<Shift>**+ **<@>** key-right next to 'P'- to get the '|' symbol)
Also,**"Auto Usb" function is by default activated,** so if you have a usb stick plugged,
USB mode is automatically activated. If not, then you must give RSX command: **|USB** after cold boot.
Finally, with **|HELP** you get a list of all available RSX commands with short descriptions.

AMSTRAD CPC 664/6128:

For direct file access:
- |**CAT** to get a catalogue of root directory
- |**CD,"dir"** to change directory to "dir"
- |**CAT** to get catalaogue of "dir"

Then you just give RUN"NAME" of the BAS/BIN file, exactly like you do when using a floppy disk!

For Amsdos dsk image access:
- |**CAT** to get a catalogue
- |**MG,"name"** to select the image file you want to use
- |**FDC**, to enable Floppy Disk Controller emulation

From now on, any CAT,LOAD,RUN,SAVE command will access dsk image!
And by giving again "|FDC" command, FDC emulation will be disabled and CAT,LOAD,RUN,SAVE commands will access again the floppy disk drive!

For Parados dsk image access:
-|PARA , Amstrad will reset with Parados Rom installed, and you can directly access any parados dsk image or floppy disk.

AMSTRAD CPC 464:

You can just give RSX command: **|6128** which instantly convert CPC464 to CPC6128 and follow the above instructions. Otherwise:

For direct file access:
- |**CAT** to get a catalogue of root directory
- a$="dirname"
- |**CD,@a$** to change directory to "dirname"
- |**CAT** to get catalaogue of subdir

Then you just give RUN"NAME" of the BAS/BIN file, exactly like you do when using a floppy disk!

For Amsdos dsk image access:
- |**CAT** to get a catalogue
- a$="imagename"
- |**MG,@a$** to select the above disk image file
- **|DOS,1** for Selecting AMSDOS Emulation
- |**464**, Amstrad will Reset, to enable Floppy Disk Controller & DOS emulation,

From now on, any CAT,LOAD,RUN,SAVE command will access dsk image!

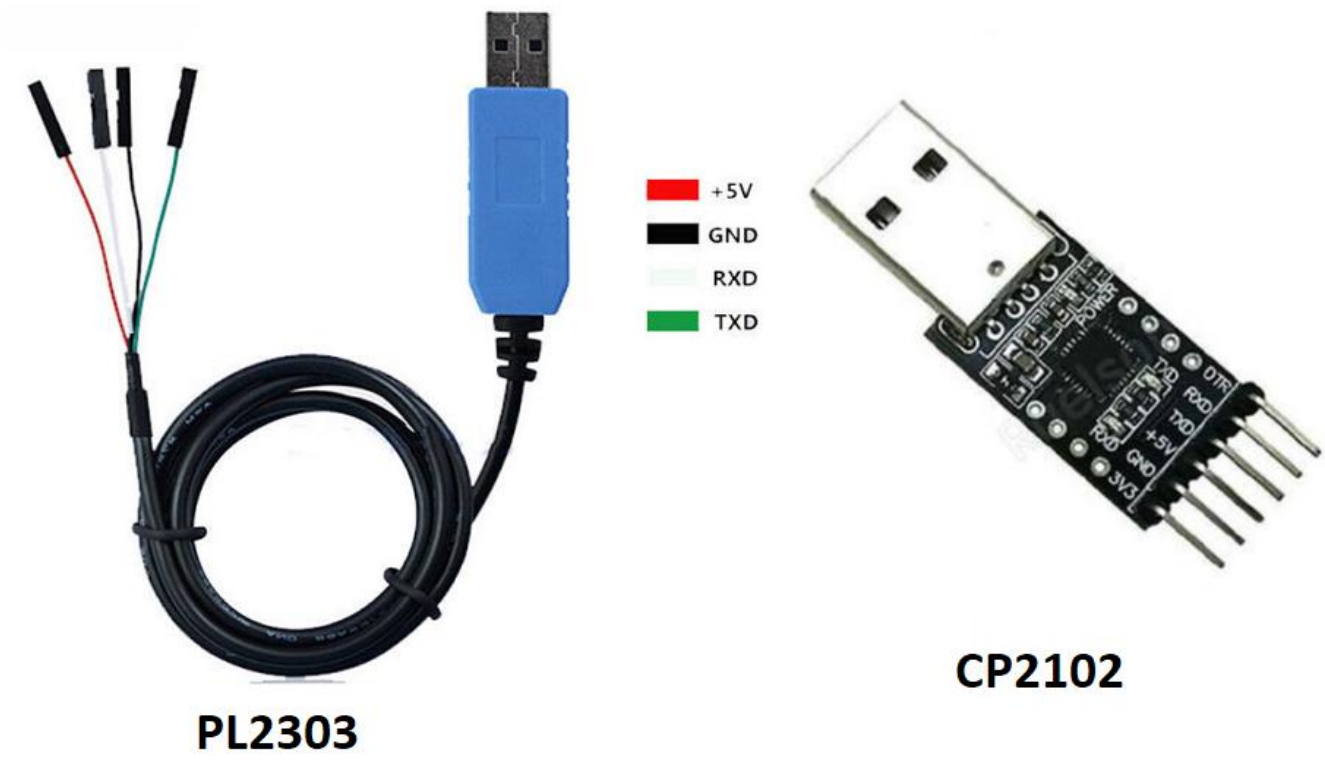
For Parados dsk image access:
Follow the above first 4-5 steps,but then give:
- **|DOS,2** for Selecting PARADOS Emulation
- |**464** Amstrad will reset, and then you can access the PARADOS dsk image using CAT,LOAD,RUN,SAVE commands.

Serial Interface Connections

Serial interface can be used with:
(* For more details about all RSX commands, check the "**RSX COMMANDS**" section bellow)

Direct serial cable:

Connect using a usb2serial cable adapter, like **this** (PL2303TA, you can find drivers **here** ,also verify which chip version you need for the windows version you use) or like **this** (CP2102,for this type you will also need 4 female to female DuPont cables). This can also used for Update/Repair ULIfAC board Firmware (check "**Updating the Firmware of ULIfAC**" section).



Connect **ONLY** Tx=Green, Rx=White, and Ground=black to serial interface pins, **red** / +5v cable/pin **MUST NOT** be connected!

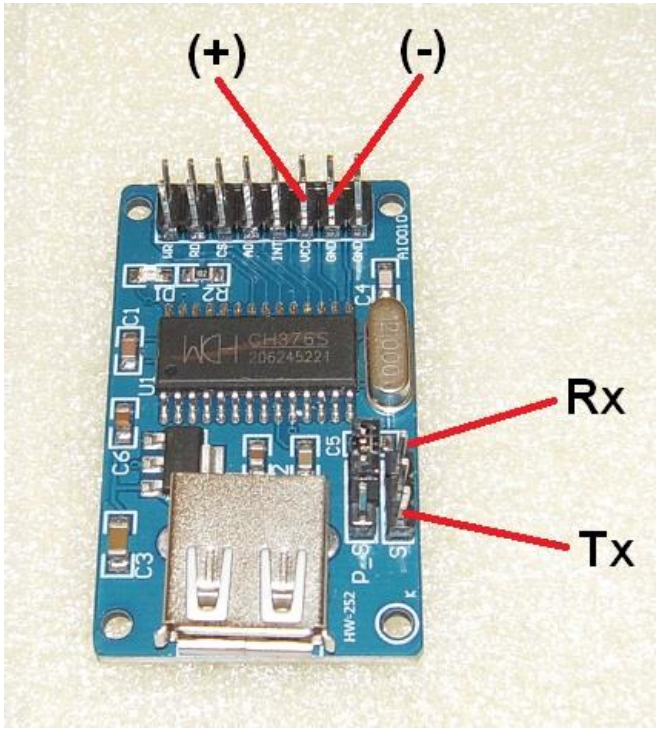
After you made the proper connections, just set serial port speed to match that on PC side, using the "**|SET**" Command, and then just give: "**|EN**" to enable Direct mode.
That's it!

From now on, you can access the hard disk drive of your PC,by browsing directories (**|CD**), receive catalogues (**|CAT**), delete files (**|DEL**) and use directly **load, run, save** and **cat** commands, exactly like you do with floppy disks. Note also the amazing loading speeds (that can be up to ~30kb/sec),where **you can load most of the games in a couple of seconds**!

Moreover, you can also select disk images (**|MG**) and then enable Floppy Disk controller emulation (**|FDC** on CPC 664/6128, **|464** on CPC 464),in order to have full read/write access on that disk image! In 2 words, you convert the HDD of your PC, to a HUGE super fast floppy disk, directly accessible by your Amstrad CPC! ;-)

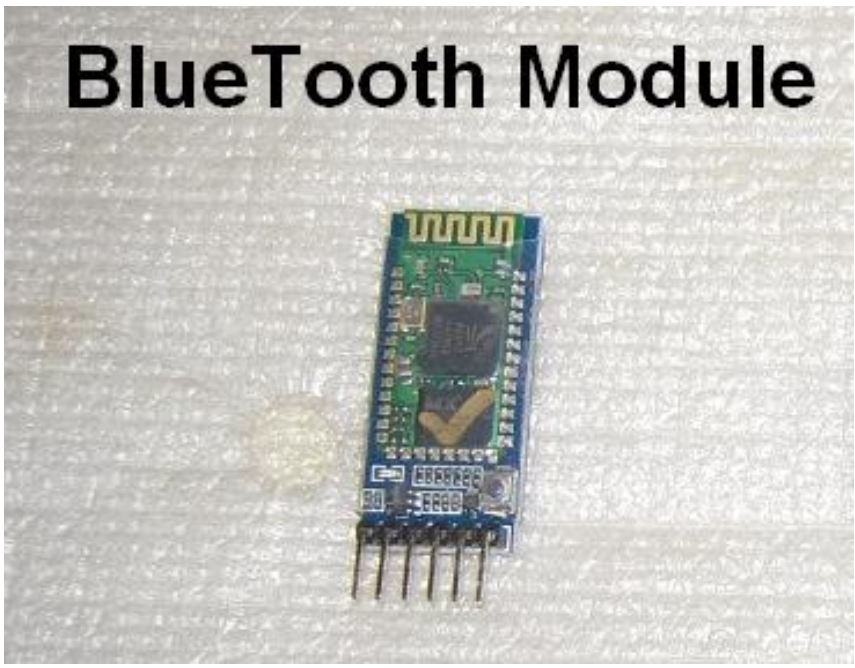
CH376 USB Host module:

This module is now mounted on the main board but you can also connect one manually: In the photo above, you can see what pins you need to connect to interface UART connector,using 4 small dupont wires:



Usb flash drive must be **FAT or FAT32** formatted. **If you experience problems in usage** (like frequent hang-ups, delayed execution of commands, erroneous messages, or even crashes),**try to create a small partition of 256Mb and format it, using FAT file sytem**. In most cases, this will solve the problems. If not, then probably you need to change the flash drive you are using.

Wireless connection using a Bluetooth module:



For direct wireless connection to a Laptop/Note book/Smart phone equipped with Bluetooth, or, two Bluetooth modules, one connected to CPC serial interface, and the other to a usb2serial cable adapter connected to a usb port. Check **"configure Bluetooth module"** on how exactly you can set up module for use. After that, everything is the same as described in previous "**direct serial cable** connection" .

Wireless connection using an ESP-8266 wifi module:

This is especially useful for creating a LAN or for direct connection to internet!
Check **"configure Esp8266 WiFi module"** on how exactly you can set up module for use. After that, everything is the same as described in previous "**direct serial cable** connection"

All of the above, are very easy to find on ebay and they cost almost nothing!(usb2serial can be found at ~1euro and other modules for 2-3euros…).

Finally, if you have an on old motherboard with **on board RS232 serial port**, you can't use it directly to connect with serial interface, but you must use an **RS232 to TTL converter** between them.

Basic Usage

Interface usage, is pretty simple and straight forward. Communication with the device is accomplished using only two ports:
&FBD1: The control port
&FBD0: The data port
And the two BASIC commands INP() for receive and OUT for send.

Send & Receive bytes:

This method can be used with direct serial/ Bluetooth module and also, WiFi module, which must be pre-configured using the **|WIFI** RSX command, in order to establish a passthrough/transparent transmission mode.

To send a byte, just give the BASIC command:
OUT &FBD0,x (x: 0-255)
Or on assembler, the instructions:
LD A,x (x: 0-255)
LD BC,&FBD0
OUT (C),A

To receive a byte, you must first check if there is any byte available at the buffer, by reading the control port. Use the BASIC command: **INP(&FBD1)**. If it returns '255', there is a byte available, if not, it returns '1'.So, in order to receive a byte from serial port, you just give the BASIC command:

IF INP(&FBD1)=255 THEN A=INP(&FBD0) (variable A contains the received byte)

And if you want to wait until a byte is available, you can use a small loop like:

WHILE INP(&FBD1)=1: WEND
A=INP(&FBD0)

And here is the assembly code for doing the same thing:

```
check_BYTE:
      LD A,&FB
      iN A,(&D1)
      DEC A
      JR Z,check_BYTE
ld      a ,&FB
in      a,(&D0)
```

Accumulator A now has the received byte.

Send & Receive bytes with a WiFi module acting as TCP server:

In order to **send data using a WiFi module** that is configured as TCP server (using the **setservr** program), you must first execute an AT command:

AT+CIPSEND=0,X

Where X, is the length of the string to be sent. Data string follows immediately after.
Below is the BASIC routine **to send a string of chars/bytes (named A$)**:

```
1000 REM SEND WiFi ROUTINE
1010 B$=STR$(LEN(A$)): C$=AT+CIPSEND=0,+RIGHT$(B$,(LEN(b$)-1))
1020 FOR k=1 TO LEN(c$)
1030 OUT &FBD0,ASC(MID$(c$,K,1))
1040 NEXT K
1050 OUT &FBD0,&D: OUT &FBD0,&A
1060 FOR k=1 TO LEN(a$)
1070 OUT &FBD0,ASC(MID$(A$,K,1))
1080 NEXT K
1090 RETURN
```

Receiving a byte/char, is done like with serial connection, BUT before getting the actual data, there is always a header in the form of:

+IPD,1,X:

Where **X** is the number of the received bytes (or length of string).
In order to **receive only the actual data**, some starting bytes should be removed:
For **1-9 bytes/chars,** first **13 bytes** should be removed or ignored
For **10-99 bytes/chars,** first **14 bytes** should be removed or ignored
For **100-999 bytes/chars,** first **15 bytes** should be removed or ignored
For **1000-2048 bytes/chars,** first **16 bytes** should be removed or ignored

Configure Interface

Default serial speed of the interface is 1065600bps.
You can give various commands to the interface, using the control port, and the BASIC command:
OUT &FBD1,x
Where **x** is:

0: For resetting the interface (without resetting Amstrad CPC)

1: Clears receive data buffer

2: Enables a special asynchronous burst mode, for fast receiving the hole receive buffer (3100 bytes) at once. When this mode is enabled, you don't use INP(&FBD1) to check if there is available byte, but instead, you receive directly data using only INP(&FBD0). When the receive buffer becomes empty, next INP(&FBD0) will cause automatically refilling of the receive buffer (e.g. 3100bytes). This mode is used in the file transfer utility, in order to avoid asking if there is a byte available for every byte to receive. **USE WITH CAUTION,** because it will freeze Amstrad, until the receive buffer is full!

3: Disables the above mode.

30: This is the command used by |STAT Rsx command, which returns the status of interface.

65: For staying to last directory used after reset.

66: For return to USB device root directory after reset.

92: Activates "Auto Usb" function,in order to automatically enable usb mode upon cold boot.

93: Deactivates "Auto Usb" function

10: Set serial speed to 300bps
11: Set serial speed to 2400bps
12: Set serial speed to 9600bps
13: Set serial speed to 19200bps
14: Set serial speed to 38400bps (useful for setting Bluetooth module)
15: Set serial speed to 57600bps
16: Set serial speed to 115200bps (the usually default speed for Wifi Modules)
17: Set serial speed to 230400bps
18: Set serial speed to 460800bps
19: Set serial speed to 576000bps
20: Set serial speed to 31250bps useful for connecting MIDI devices
21: Set serial speed to 921600 (max speed using wireless Bluetooth modules)
22: Set serial speed to 1000000bps(connection speed for usb storage device)
23: Set serial speed to 1065600bps(max speed for serial cable/Wifi module)
These are meant to be used mostly in programs, as you can set serial speed much more easily, using the "**|SET**" RSX command.

Finally,in order to **change serial interface ROM numbe**r you can give: **OUT &FBD2,x**
Where **x** is the desired ROM number (Default number is 5). This might be useful if you have other I/O devices connected to your Amstrad that might use the same ROM number.
For CPC464 you should use a number less of 7, and for CPC6128, a number less or equal of 15. Note also, that you should avoid numbers 0 and 7, because they are reserved by Amstrad CPC for BASIC and AMSDOS ROMs respectively. The new ROM number will be saved into microcontroller EEPROM memory (except for numbers '0' and '7'),and will be used from now on, so you don't need to give the command every time. Finally you can check the current ROM number, using |STAT RSX command.

Utilizing ROM Board/Dual Mode of Operation

As already noted, default operation of the board is in 512Kb Ram expansion mode. In order to use the 32X Rom Board mode or Dual 256Kb ram+ 16X Rom board mode, first you need to mount Rom files into desired rom slots, using the simple **|MROM** Rsx command. Note that rom mounting is rather fast, each 16K Rom takes ~half a second to mount into board's 512K SRAM (the part that acts as Rom board). Then you can select mode of operation (32X Rom board or Dual mode), enable/disable Rom slots and activate lower rom, using **ROM configuration utility**.

You can also use simple I/O (OUT & INP) commands for manually or in-code configuring:
(Note: all of the commands bellow, can be used in 512k RAM expansion Mode or 32X Rom mode, Dual mode is not supported):

Setting Rom mode:

- **OUT &EF00,71** :Set 32X ROM mode
- **OUT &EF00,72** :Set Dual 256K Ram expansion + 16X ROM board mode
- **OUT &EF00,70** :Set CPR mode (experimental)

Board will remember your choice. Giving **|SW** or pressing RAM/ROM button, will switch to selected Rom mode. Giving **|SW** or pressing RAM/ROM button again, will switch to default 512K RAM expansion mode.

Configuring Roms:

OUT &EF00,60 :Disable all upper rom slots
OUT &EF00,61 :Enable all upper rom slots
OUT &EF00,X : If 'X' is 0-31 it enables Rom slot 'X'.
If 'X' is 100-131 it disables Rom Slot 'X'.
If 'X' is between 151-181, it sets mapping(X-150 e.g. 1-31) of upper rom 0,when lower rom is enabled.
OUT &EF00,50 :Disables Low rom
OUT &EF00,51 :Enables Low rom

Number 'X' can be 0-31 for 32X ROM mode, or 0-15 for Dual 256k/16X ROM mode.

Using the above commands you can easily **create your own rom configuration Basic programs** and save them to the usb stick, in order to instantly reconfigure your Amstrad with the desired rom configuration scheme after each cold boot! For example:

```
5   OUT &EF00,72:REM selects Dual mode
10  a$="ROM_NAME1.ROM":num=0
20  |MROM,@a$,num: REM mount ROM_NAME1 to rom slot 0
30  a$="ROM_NAME2.ROM":num=2
40  |MROM,@a$,num: REM mount ROM_NAME2 to rom slot 2
50 OUT &EF00,51:REM activates Lower rom
60 OUT &EF00,102:REM disable upper rom 2 (which is auto-enabled with |MROM in line 40)
70 OUT &EF00,0:REM enable upper rom 0
80 OUT &EF00,152:REM maps rom slot 2 to upper rom 0
90 |SW:REM reset Amstrad to Dual mode (alternatively press RAM/ROM button).
```

With the above program, we install a lower rom and an upper rom 0 (which is mapped to rom slot 2, as slot 0 has the lower rom)

Other useful commands:

OUT &EF00,X:A=INP(&EF00) :X should be between 200-231, 'A' will return the state of selected rom slot (0=disable,170=enabled)
OUT &EF00,195:A=INP(&EF00) : A will return the mapping for upper rom 0 when low rom is enabled
OUT &EF00,194:A=INP(&EF00) A returns current ram expansion state: 0=disabled, 100=enabled
OUT &EF00,196:A=INP(&EF00) A returns 1 for 512K RAM expansion mode, or 2 for 32X ROM mode.
OUT &EF00,199:A=INP(&EF00) A returns low rom state: 100=disabled 150=enabled.
OUT &EF00,191:A=INP(&EF00) A returns mapping for rom 0 in dual mode
OUT &EF00,192:A=INP(&EF00) A returns mapping for rom 1 in dual mode
OUT &EF00,40 :Disables Ram expansion
OUT &EF00,41 :Enables Ram expansion

RSX Commands

- **|HELP**: Shows all RSX commands with short description and usage. With optional argument (**|HELP,2 |HELP,3**) shows directly 2nd or 3rd help page.

- **|CAT,("X")**: Without argument, shows catalogue of files and sub dirs.With optional argument "X" , it will show files/Folders, starting from 'X'. In usb device mode, you can use more than one letters for filtering. Finally, with argument '1'(|CAT,1) it will give a list of DSK images in selected dir (PC Direct Mode only).

- **|CD,"name"**: Change dir to 'name'. Without argument, moves up a dir.

- **|CDR**: Moves up to root directory.Use with USB Device only.

- **|USB**: Enable /Disable CH376 Usb Host Module,to access USB Storage Device. You need to give this command **only once** after initial powering Amstrad. Board will "remember" the last state of the usb device, and the dsk image(s) selected.

- **|EN**: Enables Direct Mode 1 to access files from PC or USB Device program (done automatically with "|USB" command).After issuing this, all **LOAD, RUN** and **CAT** commands, will access files (or have a catalogue) from PC or USB storage device. **Direct Mode**,must be active before load/run/save a game/program.

- **|EN2**: Enables Direct Mode 2,try this,if above Mode 1 fails to load a game/program (NOT to use with USB device)

- **|FDC**: Enables/Disables 765 Floppy Disk Controller (765FDC) Emulation. This command is used for accessing DSK images on Amstrad CPC 664/6128.

- |FM: Loads file manager.

- |FSM: Loads file size manager (same as above, but it also shows size of each file, a bit slower than File manager)

- |6128: Loads CPC 6128 Firmware ROM, Basic 1.1 ROM, AMSDOS/PARADOS Rom (depending on the emulation mode selection, using |DOS RSX command), and resets Amstrad to DUAL RAM/ROM mode. **This command practically converts any CPC464 to CPC6128 with 256Kb extra RAM!**

- |SW: Switch between RAM mode and ROM Board/Dual mode. You can use this, to swap modes alternatively to RAM/ROM button. Doesn't function in Dual mode.

- |PARA: Loads parados Rom and reset Amstrad to dual RAM/ROM mode. Use this instead of |464 if you want to have faster access of large parados dsk images.

- |MROM,"filename",X: Loads a rom from usb stick to rom slot X and activate it (e.g. after reset in 32X ROM mode or DUAL RAM/ROM mode the specific rom slot will be initialized). 'X' can be 0-31 for 32X ROM BOARD mode, or 0-15 for dual RAM/ROM mode

- |ROM: Loads the configuration utility for setting up rom mode, roms, enable lower rom etc.

- |TYPE,"file": Displays on screen ascii/text files. For larger texts, use SPACE to pause printing.

- |DIA: Loads the Amstrad CPC diagnostic utility where you can check & test RAM/ROMS/KEYBOARD/JOY etc

- **|DIS**: Disables Above Direct and FDC Modes,in order to Access Floppy Disk Drive. After that, every **LOAD, RUN, CAT** commands, will access Disk Drive again. Alternatively, you can give: **OUT &FBD1,4**.The latter is very useful, if you happen to get “unknown command” error message, when trying to use an RSX command.

- **|464**: Resets Amstrad and enables Amsdos/Parados and 765FDC emulation. This is an obsolete command but I kept it for backward compatibility.

- **|MD,'NAME'**: Create directory 'name' in current Directory of a Usb storage device.(Usb Device Only).

- **|DEL,'XXX'**: Delete all files starting from 'xxx',from selected directory. With PC direct mode,you can also use wild card '*' for multiple file deletion. Of course you can also give a single file name as argument, to delete this file only.

- **|DELD,'name'**: Delete the directory 'name' from Usb Storage Device.

- **|DOS,x**: Select DOS Emulation,x=1 for Amsdos or x=2 for Parados.This is used in order to determine which DOS ROM will be used with DIRECT mode, and also, after “|464” command is given.

- **|COPY,'xxx'**: Copies from USB Device to a Floppy disk,all files starting from 'xxx'. You can also use an exact file name as argument, to copy a single file only.

- **|UCOPY,"name1","folder path",("name2"):** Copy a file into usb device, named "name1", from it's current position, into another folder. Optionally you can give "Name2" to copy file with a new name.For example: **|UCOPY,"GAME.BAS","/DIR1/GAMES","GAME2.BAS"** ,will copy ‘GAME.BAS’ file to ‘/DIR1/GAMES’ subfolder, with name "GAME2.BAS". If you omit the last argument, file will be copied with same name.

- **|MG,”name”**: Mounts DSK image 'name' and place it to 1st image slot. You can access this disk image file, by using “|*FDC*” 765FDC emulation. When in PC DIRECT mode, you can also use it without argument, to enable automatic selection of image,e.g. 1st image found in current PC directory, will be selected (“Auto image selection” check box, will be checked in PC windows utility).

- **|MG2,”name”**: Mounts DSK image 'name' and place it to 2nd image slot. This is required for loading multiple disk image games/programs.

- **|MG3,”name”**: Mounts DSK image 'name' and place it to 3rd image slot.

- **|MG4,”name”**: Mounts DSK image 'name' and place it to 4th image slot.

- **|SMG**: Shows a list of the image files mounted on all 4 image slots, and which slot is currently selected. You can round select image slots by pressing “dsk swap” button.

- **|SET**: A small utility for easy setting Serial port speed

- **|WIFI**: BASIC utility for easy configure ESP8266 WiFi modules.

- **|DSK,”name”,x**: Use this command to copy DSK image ‘name’, from usb device to a floppy disk. Optionally, you can use the 2nd argument ‘x’ for Drive A, or ‘2’ for Drive B. Without argument, is used for copying a selected image from PC to Floppy disk.

- **|DISK,name,(1):** Creates a dsk image file from Floppy disk. Supports DATA/SYSTEM/IBM formats and up to 42 tracks. The optional 2nd argument '1', is for auto creating new image filenames, by adding a letter at the beginning of the name. For example,if you give name: "BACKUP.DSK" ,with extra

argument '1', name images will be: "ABACKUP.DSK", "BBACKUP.DSK","CBACKUP.DSK" etc. This way,you can take many backups in a row, by only changing disks and pressing a button.

- **|SNA,addr**: Loads SNA snapshot files. Optionally you can use argument 'addr', (to place the required final code in the procedure), to specific 'addr' address. This is useful, for snapshots files, that can't be loaded correctly, using the default address. You can use **CPC loader creator** windows utility, to automatically create a Basic loader file for snapshot files.

- **|FORMAT,"D/S","A/B"**: Utility for easy and quick formatting floppy disks in just 15seconds! Without arguments (|FORMAT) it will directly format a disk using Data Format in Drive A. For System format and/or use of Drive "B" you must give extra arguments, for example: **|FORMAT,"S","B"** will use System format on Drive B.

- **|EXT,"filename":** Extract a file from a dsk image to USB storage device.

- **|2PC**: Utility for copying files,from Amstrad CPC floppy disk,to selected directory on PC or USB Device.

- **|2CPC**: Utility for copying files,from PC to Amstrad CPC disks.This is the equivalent of the above "|COPY" command, but for use with PC Direct mode.

- **|TER**: Terminal Utility.

- **|COM**: Small Terminal for easy communication with Wifi/Bluetooth Modules.

- **|WROM,'name'**: Write to Microcontroller's EEPROM,first 1022 bytes of 'name' file.

- **|WROM,adr,size**: Write to EPPROM,'size' bytes,starting from RAM address 'adr' - (Size can't be more than 1022 bytes)

- **|LROM,addr**: Loads EEPROM contents to RAM,at 'addr' address.Note also, that, you can use an **RST3,&FC00 assembly command**, in order to directly execute any code routine you have uploaded to EEPROM.(last 1kb of selected upper rom @ &fc000-&fffd is reserved for EEPROM, last 2 bytes are reserved for saving serial port speed and ROM number, thus you have 1022 instead of 1024 bytes available ☺)

- **|STAT**: Shows current status of ULIfAC Interface (UART speed,ROM number,Status for: return to Usb device root directory, Direct Mode,FDC emulation, ROM DOS emulation)

- **|MAN**: This resembles **|CAT**,but instead of getting the list of files/subfolders of a usb device to screen,it outputs the list of files/subfolders into RAM, starting from **&2b00** , and uses '1' as name separator,'2' to mark end, e.g.: 1NAME1NAME1NAME...1NAME2.
Function is used by filemanager utility, but it would also be useful for anyone want to develop it's own programs.

- **|GALA, |PACMAN, |KGOR**: Loads GALACHIP,PACMAN,KILLER GORILLA game respectively.

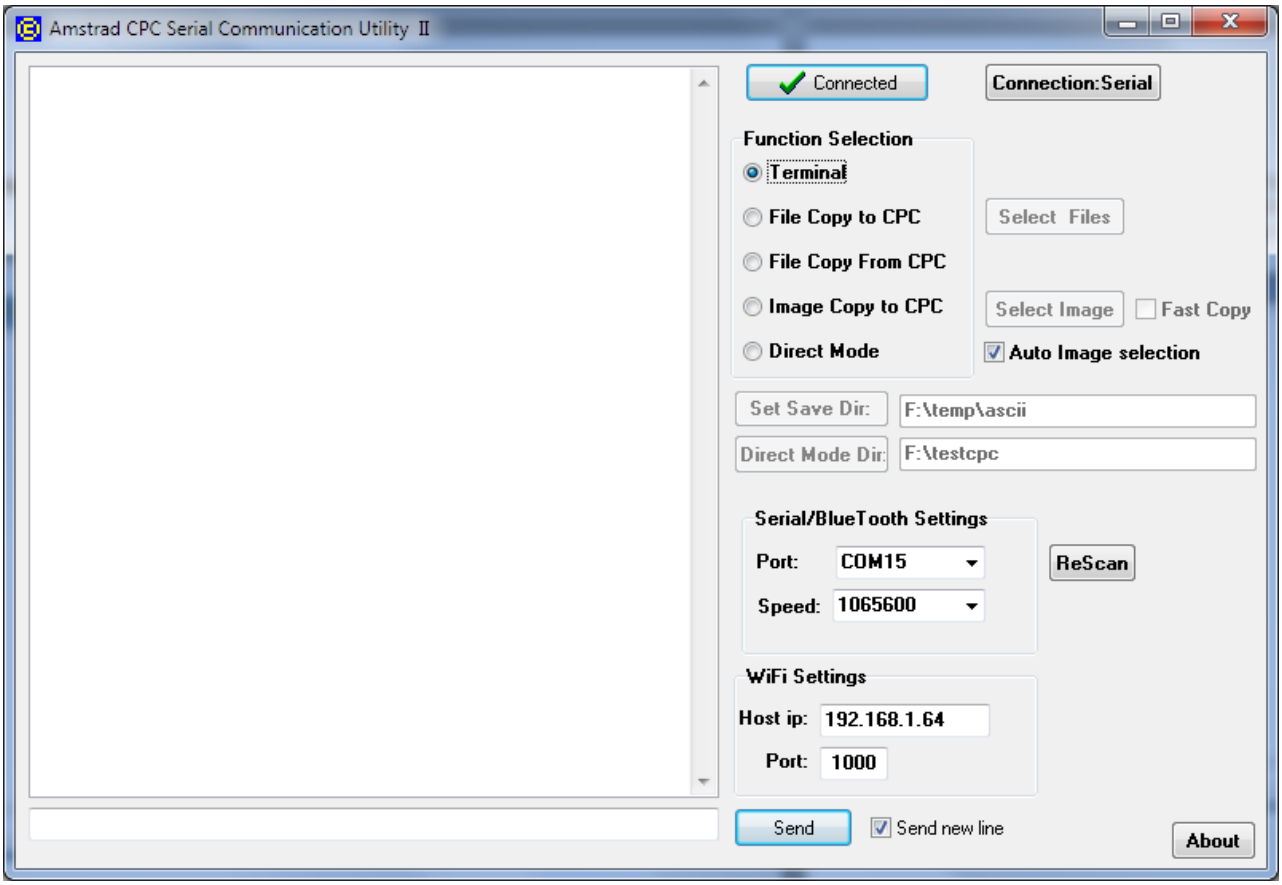
Finally you can use:

- **OUT &FBD1,92** to enable **automatic usb device activation** upon boot, and **OUT &FBD1,93** to disable

- **OUT &FBD1,66** to enable **automatic "return to root directory"** after reset, and **OUT &FBD1,65** to disable.

Windows PC Communication Utility

I have developed a utility for windows, that allows communication with Amstrad CPC:

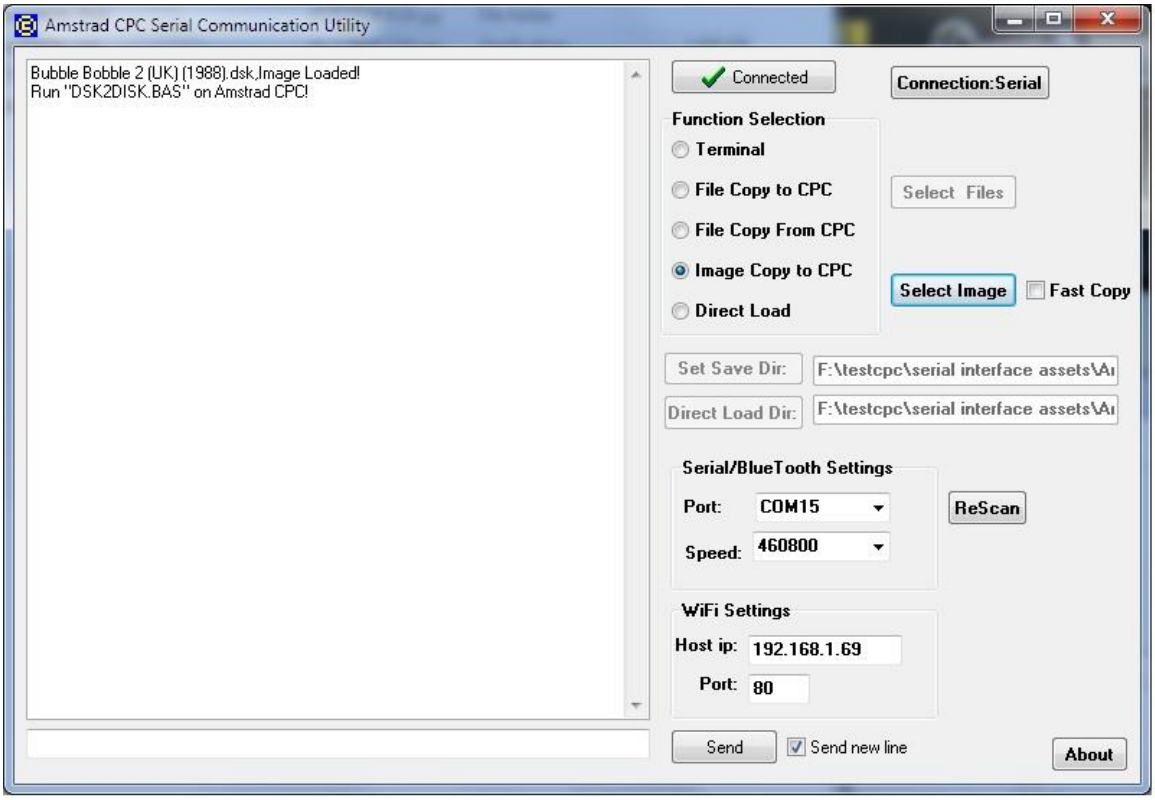


It offers five functions:

- **Terminal:** A terminal for direct communication through serial/Bluetooth or Wifi modules.
- **File Copy to CPC**: A File Copy utility from PC directly to Amstrad CPC disk.
- **File Copy From CPC:** A file Copy utility from Amstrad CPC disk to PC.
- **Image Copy to CPC:** A dsk image copy utility from PC to Amstrad CPC disks.
- **Direct Mode:** Select this function for accessing files directly from your Amstrad,using either |EN or |EN2 RSX commands.

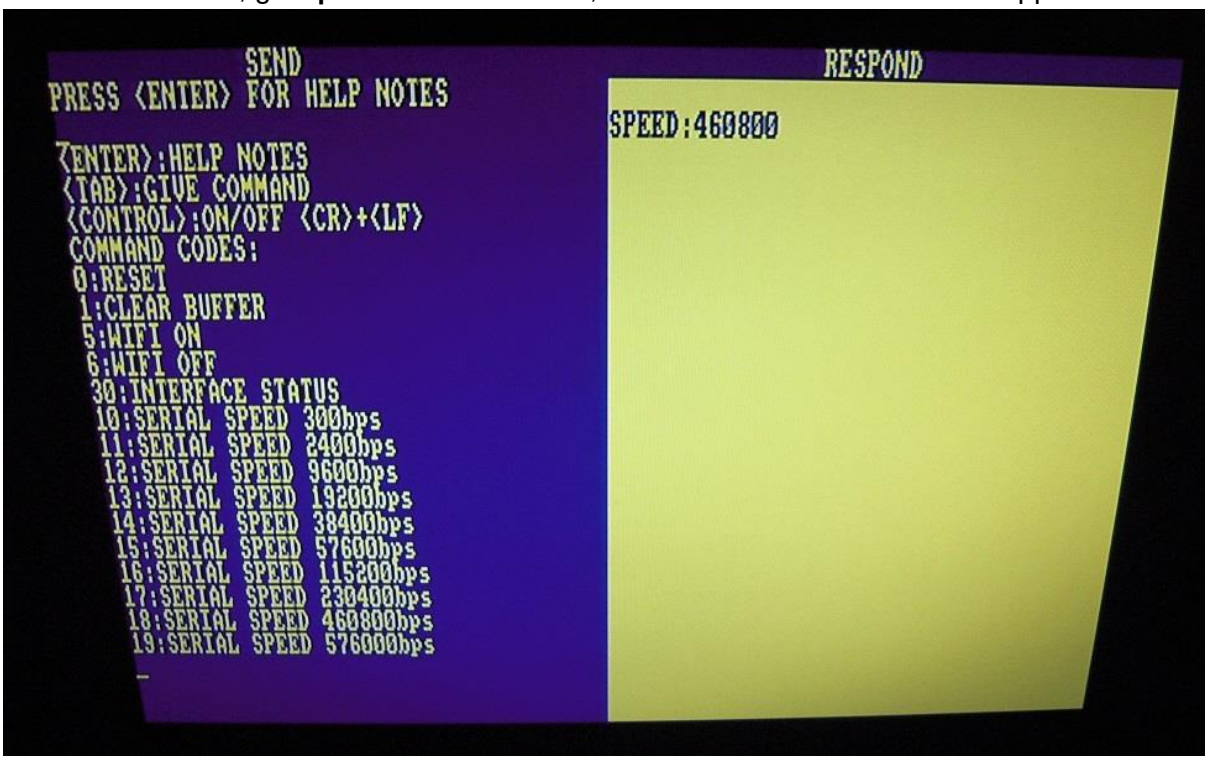
For more details on the above functions, see next sections.
Now,in order to connect with serial interface on Amstrad CPC, you must first setup your connection.This needs to be done only the first time, program will remember your settings, and set them automatically every time you run the utility.
First, using the **connection** button on the right top, select the type of your connection (Serial or WiFi), according to what you are using.
Then,from groupbox **Serial/Bluetooth Settings** you can select Com port (note: **Rescan** button, allows you to rescan for serial com ports, in case you enable/disable Com ports after running the utility), and speed for Serial/Bluetooth connection. If you use a WiFi module, connection is established from AMSTRAD CPC (see section: configure Esp 8266 WiFi module). Make sure that Serial interface on Amstrad CPC has the same speed settings and, if you are using Bluetooth module, the module is properly set (see Configure Bluetooth Module section, on how exactly to do that).

Press **Disconnected** button, it should turn to **Connected:**



Terminal Function

On Amstrad CPC, give **|TER** RSX command,and the main terminal window appears:



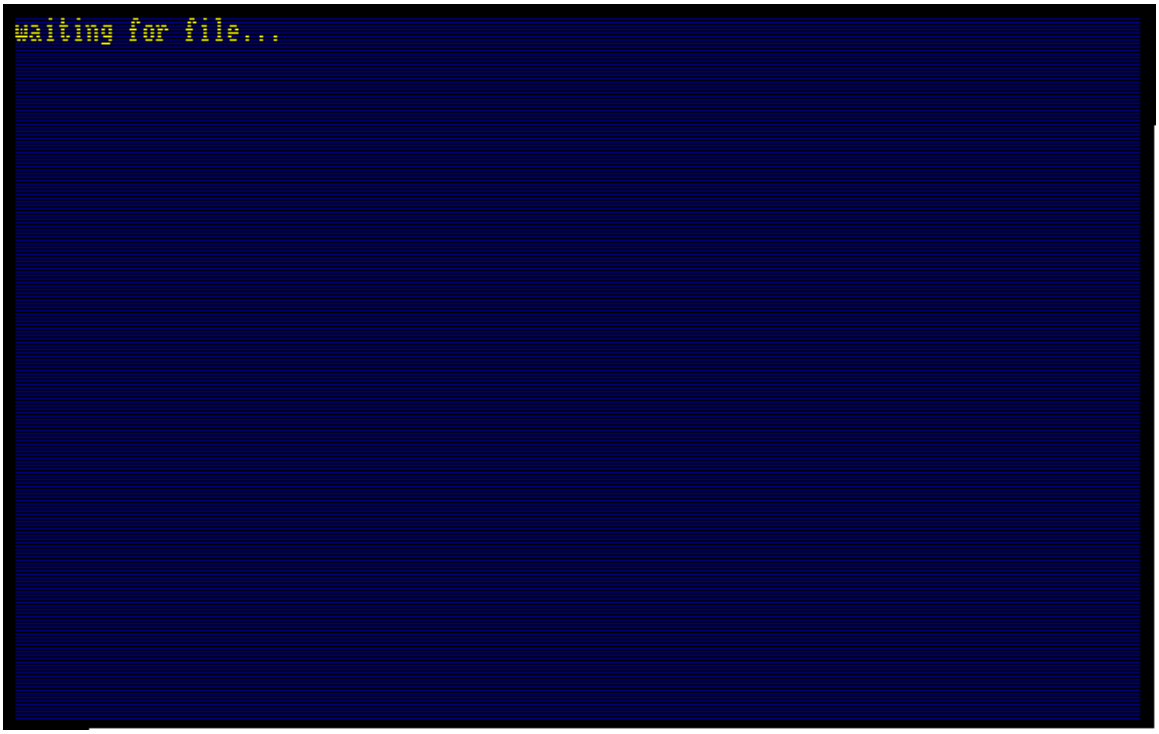
On the left side, write anything, and by pressing <Return> it will be transmitted to PC. On the right side you will receive any respond from the PC.

With terminal, you can also communicate with serial interface by sending various commands to change port speed, clear receive buffer, get status etc. By pressing <Enter> (or <COPY> on WiFi mode), you will get a list of all available commands. Commands are sent by pressing <Tab> and then the desired code.

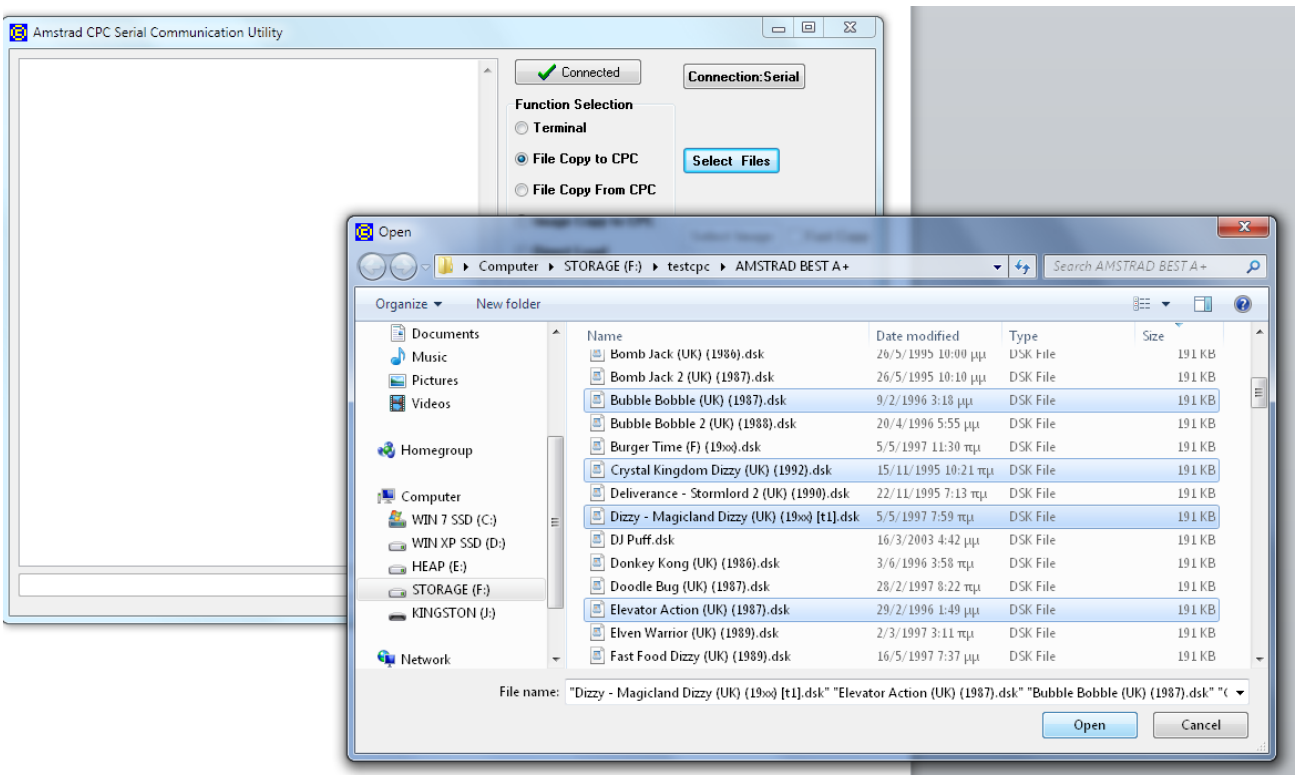
On PC side, after setting and enabling connection, select Terminal function. You can use the edit box on the left of the <Send> button to write a message and by pressing Send button it will be transmitted to Amstrad CPC.

File Copy to CPC Function

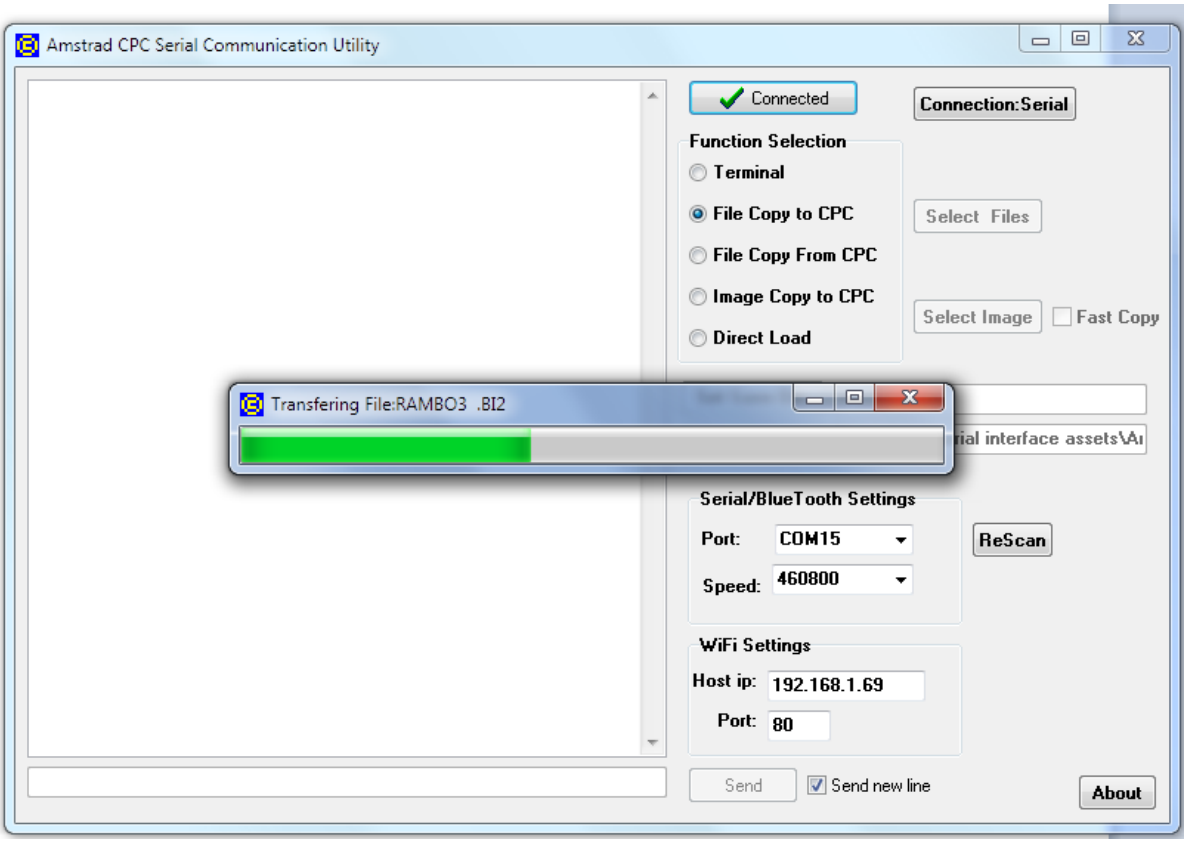
On Amstrad CPC, give **|2CPC** RSX command. You need to set the serial port speed (for WiFi mode, don't forget to use **|WIFI**, to establish the connection) , and then, you should get a waiting for file… message on screen:



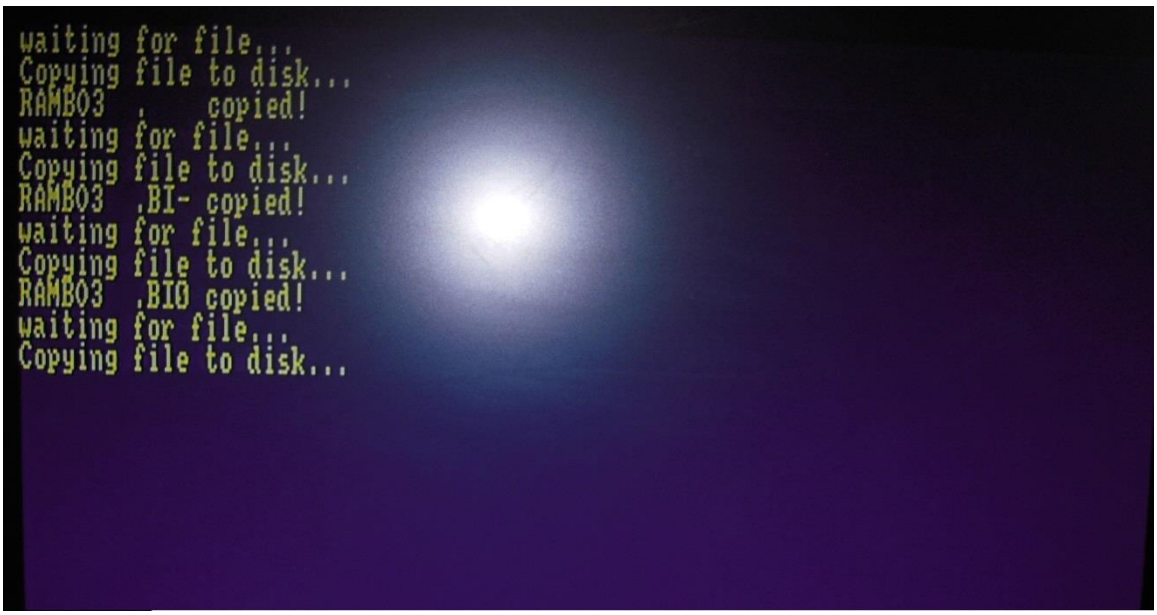
Insert a formatted disk into Disk drive.
Run windows utility on the PC.After setting and enabling connection, select **File Copy to CPC** function. Then, press Select Files button and select one or more files to transfer:



Press Open Button and file copy will begin:



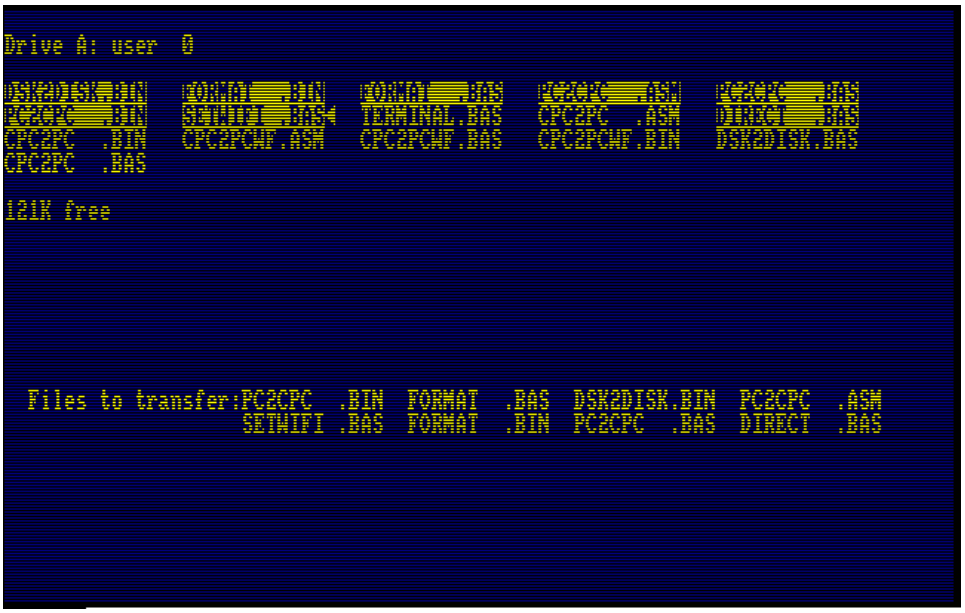
On Amstrad, you will get a confirmation for each written file:



After finishing transferring the files you want, you can reset Amstrad to exit the program.

File Copy From CPC Function

On PC, after setting and enabling connection, select **File copy From CPC** function.
Use the **Set Save Dir** button, to set the save directory for the files transferred from CPC. On Amstrad CPC, give **|2PC RSX command**. You will get a files catalogue of the inserted floppy disk:

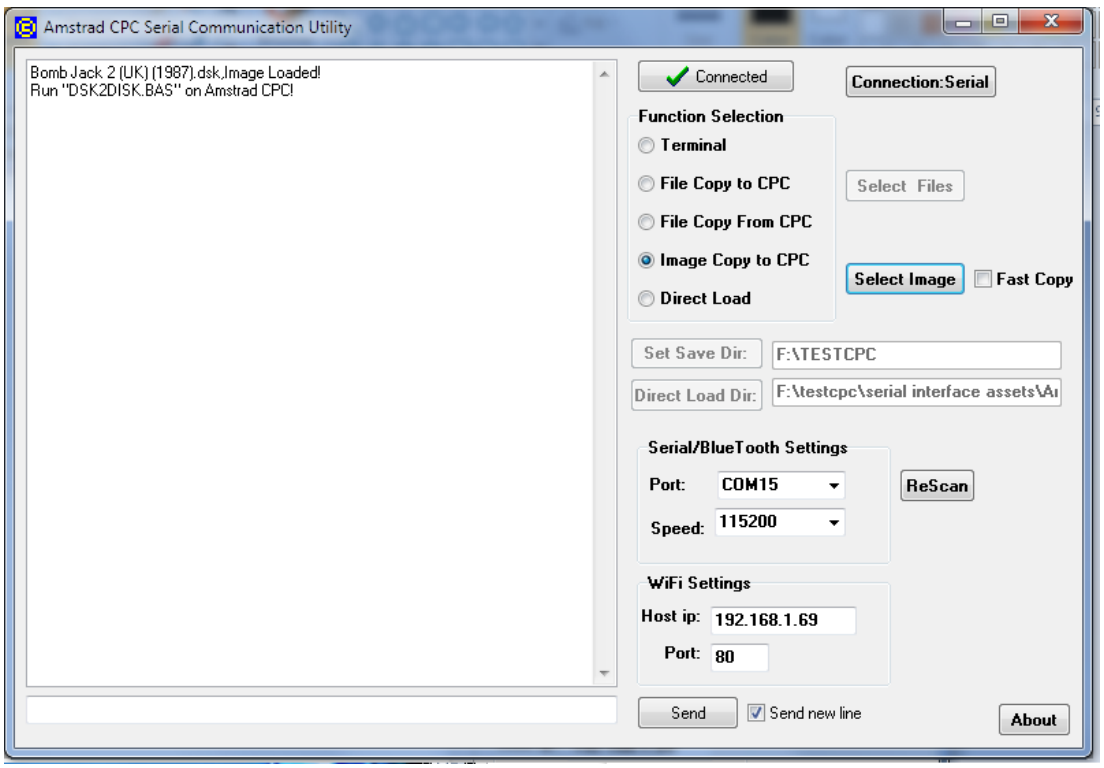


Use **arrows** to move between files, **<space>** to select/unselect a file, and **<Return>** to execute transfers of the selected files. By pressing **<1>**, you will receive DRIVE A file catalogue and by **<2>** , DRIVE B file catalogue.
Finally, with **<Caps Lock>** you can select/unselect all the files. This is useful if you want to transfer all files of the disk, or if you want to invert the already selected files (e.g. all selected files will be unselected, and all other, will be selected).

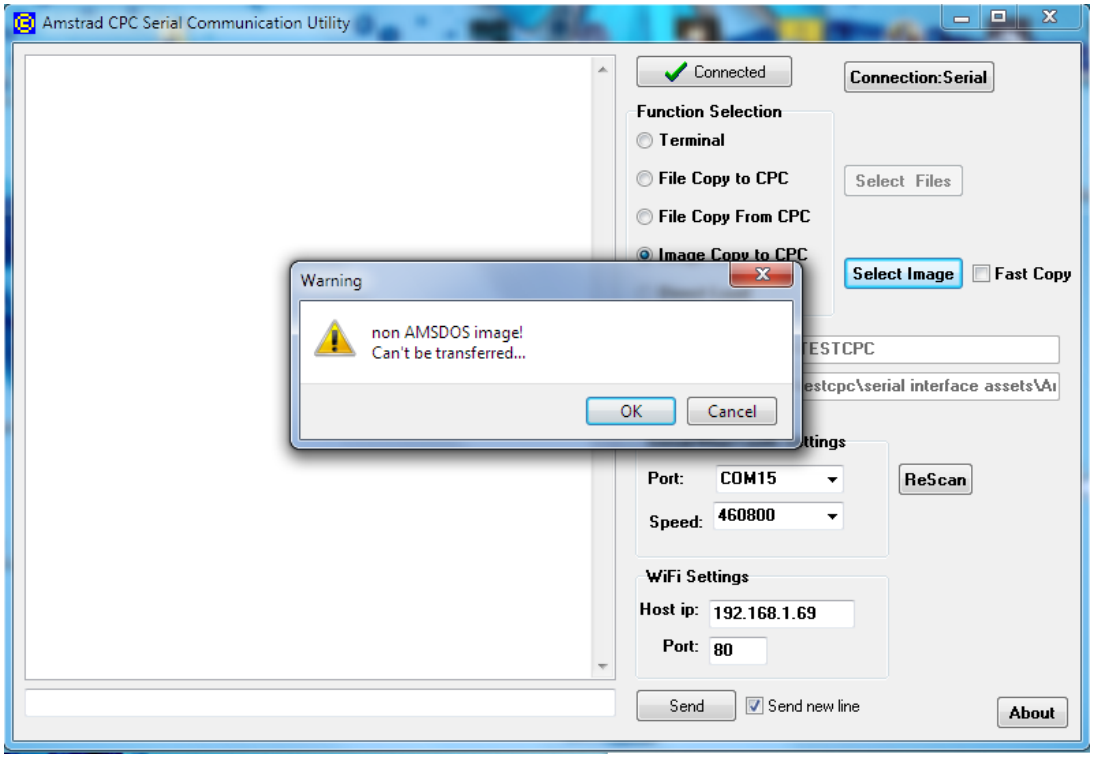
On the half bottom of the screen, you will get a list of all the selected files. Files from this list, will be removed one by one, during the transfer process:

Image Copy Function

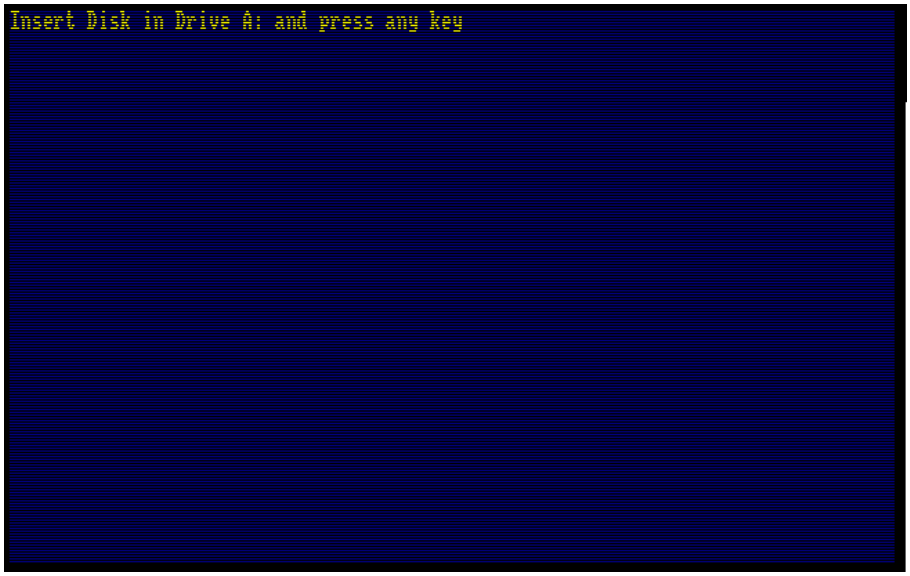
On PC utility program, select **Image copy to CPC** function. Press **Select Image** button to choose a dsk image. You will get a confirmation that image is loaded successfully:



If you select a non Amsdos image (usually copy protected games) you will by notified that currently these images are not supported:

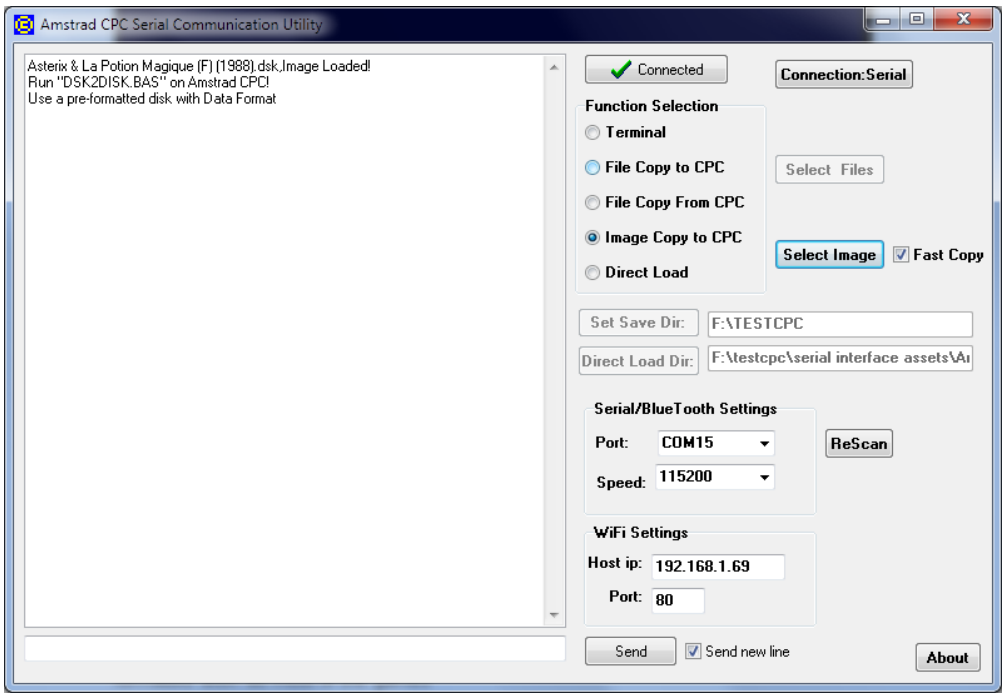


Now,give: **|DSK** RSX command on the CPC, and you will get a message to insert a disk:



By pressing any key, the transfer of the selected image will begin. Using serial speed of 460800bps or more, it should take less than a minute (~55-56seconds) to transfer a hole 180Kb disk image into any disk (either CPC/ PC/Apple formatted or unformatted).

Now, as you might already noticed, there is also a **Fast Copy** check box, right next to Select image button. By checking this, **you can transfer an image to an already Amstrad Formatted disk in just ~40 seconds**! If you check Fast Copy, you will be notified, for the format type needed (in most cases is Data Format) on the disk:

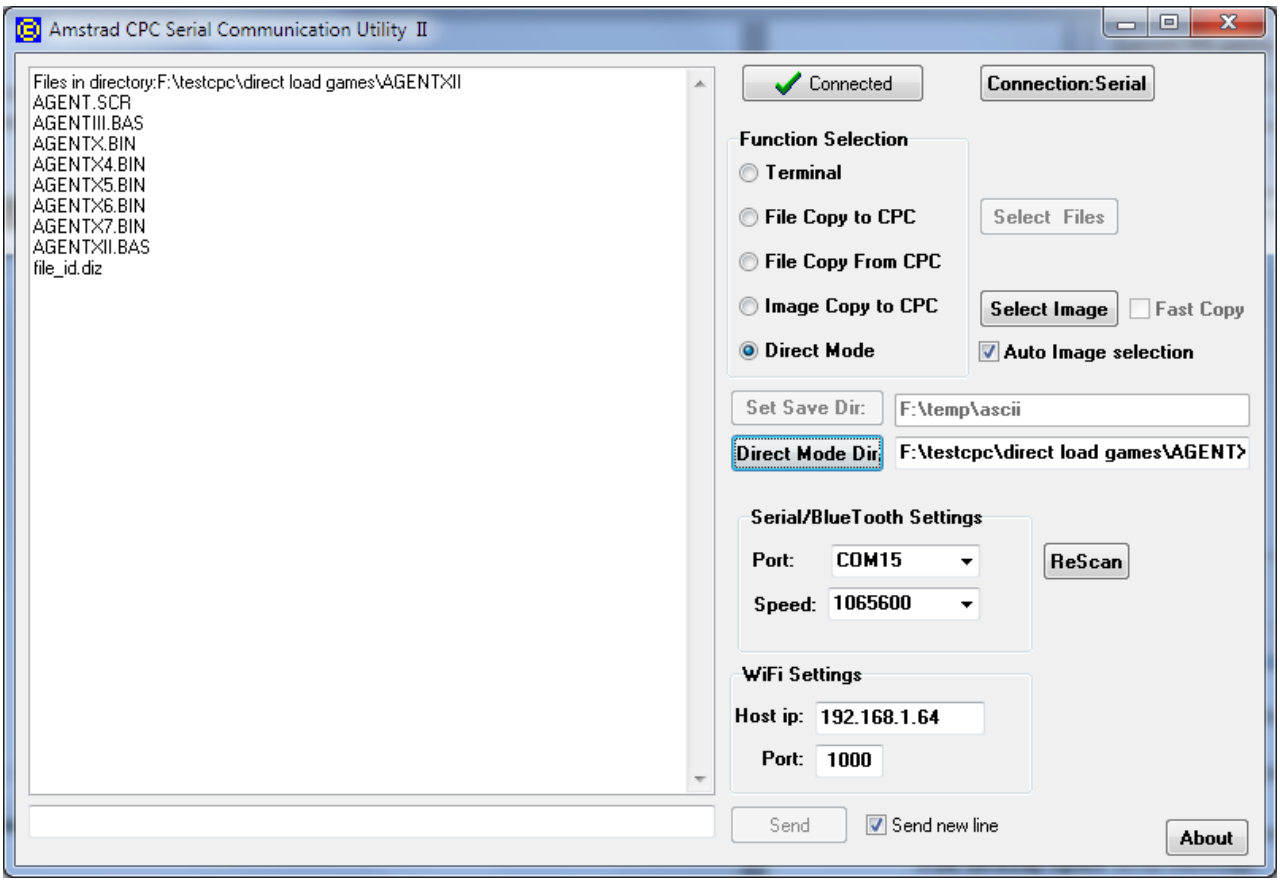


This can be proved very useful if you already transfer a game image to a disk, and you want to transfer another image on the same disk.

Direct Load Function

By selecting the Direct Load function, you practically convert the HDD of your PC to a huge storage device for your Amstrad! Through RSX commands you can **load or run a program/game (at INCREDIBLY fast speeds, of ~23-24kb/sec, more than 5 times the actual speed of a 3" disk drive!) , save files, delete files, change directories, even access DSK images (at slightly faster speed than with a real disk drive)!**

By giving a simple |EN or |EN2 to your Amstrad, any **LOAD, RUN or CAT** command you will give, it will be redirected to PC! Using button **Direct Mode Dir**, you can manually set the loading directory and have a list of all available files in it:



You can also use |CD,"name" command to change directory (or |CD to move up a directory), directly from you Amstrad!

About ~**90% of games can be directly executed using this super fast direct mode.** For the rest, you can always use the dsk image file of the game(although in that case, access is much slower, usually slightly faster than with a real disk drive/gotek drive).

You can find a list of many '**Classic & Worth to Play**' games (almost all in file form for fast access) **here**. And **here** is a ready made list of Basic Loaders for easy loading of all games in the list (check **Amstrad CPC loader creator** bellow).

Finally, a couple of useful notes about Direct Load function usage:

- If you try to run/load a file that it doesn't exist, or there is an error in loading, Amstrad will give you a "**File already open**" error message.

- For loading files with .BIN extension, this can be omitted (e.g. you can type: **RUNGAME**, instead of: **RUNGAME.BIN**, regardless of the file having the BIN extension or not). But you must include extension to load any .BAS files.

- If you have any problem loading a game/program, and reseting Amstrad doesn't resolve it, try to turn off/on Amstrad, and if this don't work, go to PC program and select momentarily another function (like terminal) and then set to Direct Load again (this will reinitialize the PC routine in case it has hang/crash).

Other Utilities & Tools

Automatic mounting of Roms upon cold booting

In order to automatically mount roms upon booting Amstrad CPC, you should place a simple text file, named: **"CONFIG.TXT", in root directory of the connected usb stick.** If such a file is found upon booting, then ULIfAC will try to load the roms declared in the file (if not,or if any error occurs, it will continue booting in 512K Ram mode as usual), and then reboot in dual RAM/ROM mode!
(note:If Amstrad CPC freezes after loading Roms, just press “Reset” button, and it will reboot in dual RAM/ROM mode, as it should)
Configuration text file structure is very simple too, just lines in quotes, containing rom name and desired rom number, for example:

"OVERKOB.ROM,4"
"CH.ROM,3"
"AMSDOS.ROM,7"
(larger than one-digit rom numbers are supported too, like 10,12,13 etc)

Low Roms are also supported, as well as upper rom 0 when Low rom is enabled at the same time. To declare a Low Rom, you place '00' after Low Rom name, and for mapping upper Rom 0 in rom slot 1 ,you should place: '01' (if you want to use upper Rom 0 without lower rom at the same time, you can also give a simple :"NAME.ROM,0")
You can **easily create such rom configuration boot files directly on Amstrad CPC** (although you can perfectly use a PC to create the configuration text file, if use notepad don’t forget to select use ANSI encoding),using a simple BASIC program:

```
10 OPENOUT"config.txt"
20 WRITE #9,"os_6128.rom,00"
30 WRITE #9,"basic11.rom,01"
40 WRITE #9,"amsdos.rom,7"
…..
90 WRITE #9,"overkob.ROM,3"
100 CLOSEOUT
```

The above program creates the required "Config.txt" ASCII file, and each line adds a rom. Note the declare of Low Rom ("os_6128.rom,**00**") and a remapped upper Rom 0 ("basic11.rom,**01**") at the same time. Of course all roms, must exist in the root dir of the usb stick too.
Here you can find ready made configuration zip files, containing the config.tx file and the corresponding 16K Rom files.
Finally, you can check any time your current rom configuration file, by giving at BASIC prompt: **|TYPE,"config.txt"** (or: **a$="config.txt":|TYPE,@a$** for CPC 464)

File manager (|FM, |FSM)

This utility can be used for easy browsing & loading files/dsk images/snapshots from usb device.
It eliminates completely the need to use RSX commands, so it would prove especially useful for CPC 464 with Basic 1.0 and the rather tiresome procedure of giving arguments to RSX commands: a$=”argument”:|RSX COMMAND,@a$

Use keys:

- **Left-Right arrows** for changing pages.

- **Up/Down arrows** for seeking file/dir names into each page.

- **<Return>** for taking action:
  -If selected name is a directory it will move into it.
  -If it's a "DSK" image it will give you a catalogue of the disk image(CPC6128/664) or it will reset (CPC464) and then you can access image by giving "CAT".
  -If it's a SNA smapshot file, it will load it.
  -If it's a file,it will try to execute it.

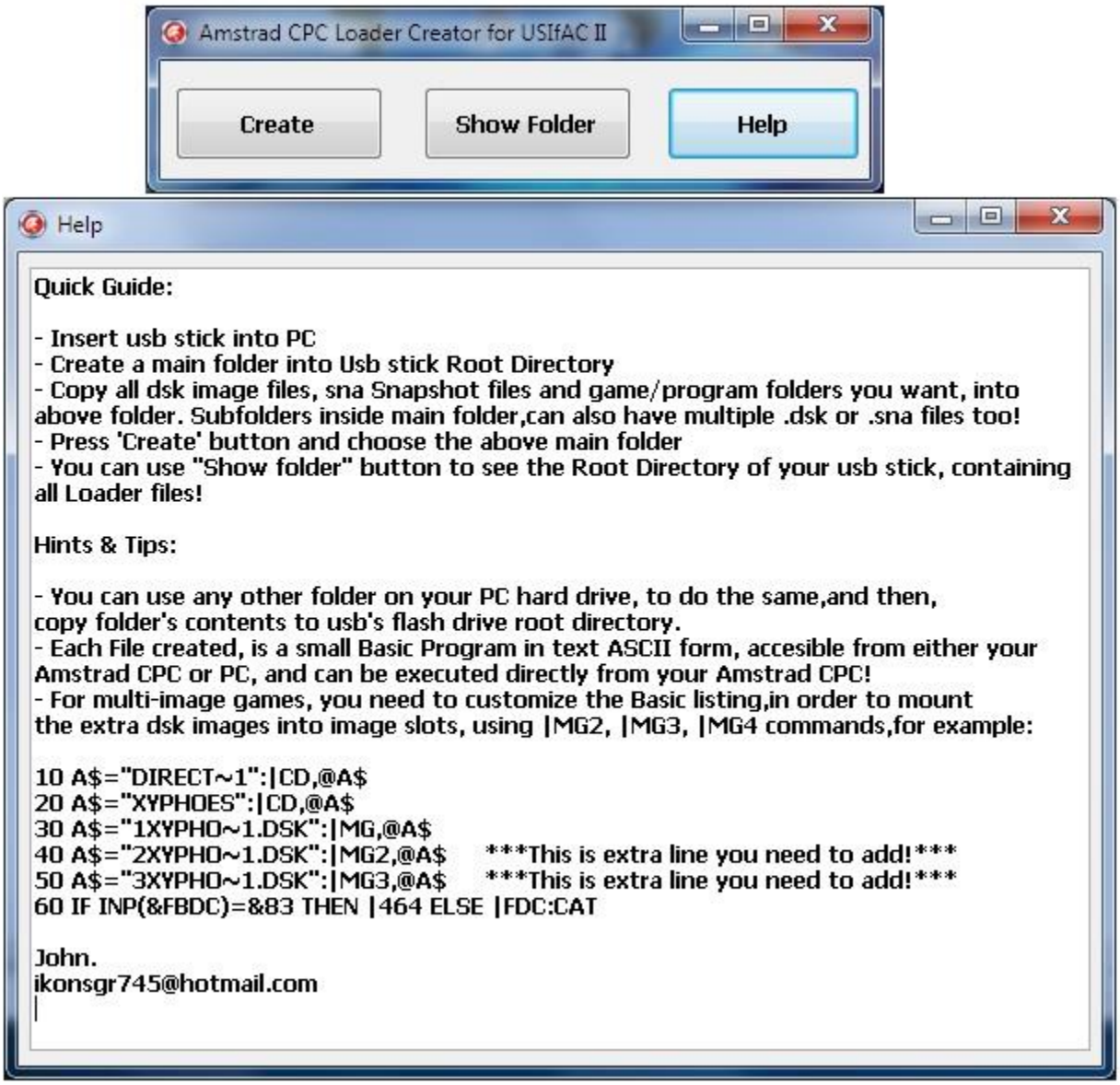
- **<SPACE>** Moves up a directory, if you are inside a sub folder. You can also use "." or ".." at the top of first page of any subfolder.

- **<1>** enables/disables treating of all names as directories. You get a note on top right corner of the screen when enabled. This is useful if some directories don't appear the <DIR> notification at the end. Alternatively you can **remove the "archive" bit property, from all files/directories, then all directories will be shown properly** (with <DIR> notification)

By continuously pressing up/down arrows you will get faster movement for quicker seeking, press momentarily for slower "row by row" seeking.
File manager **supports up to ~1000 filenames/directory and maximum 50pages** (25 names/page)
Finally**, if a game/program doesn't run directly through file manager**, just reset Amstrad, give: ‘cat’ and run game/program from "clean start" (btw, make sure that "Return to root directory" is disabled by giving an: **OUT &FBD1,65**)

Amstrad CPC loader Creator

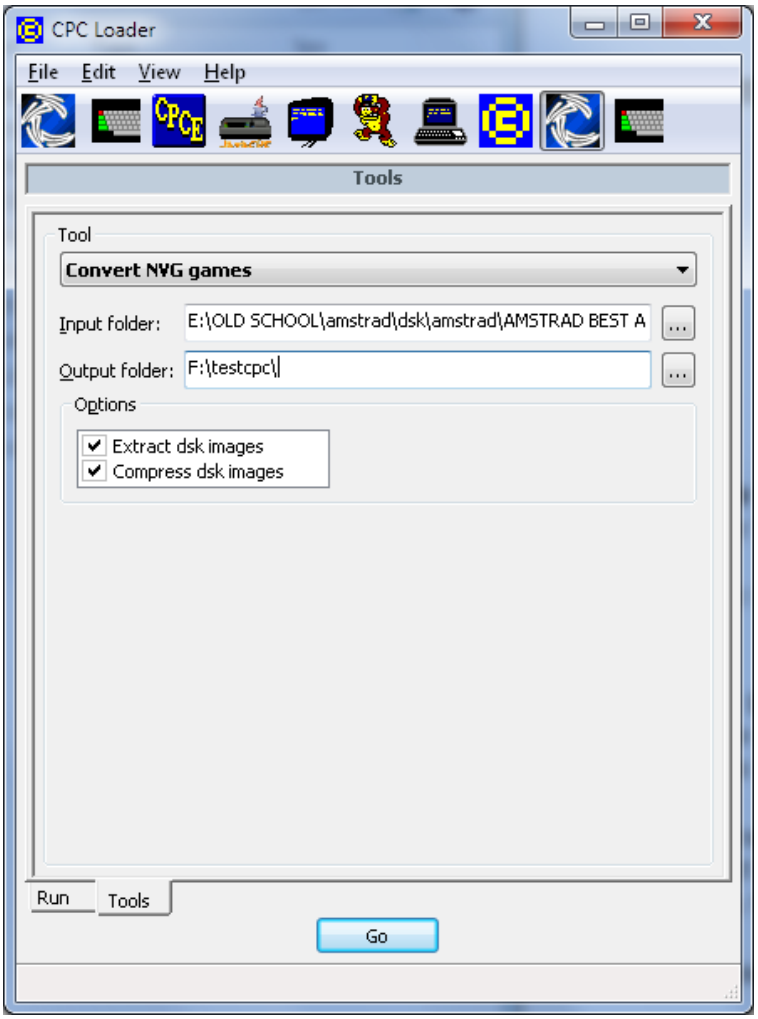
This windows PC utility (**cpc_loader.zip** can be found **here**), creates small Basic loaders (.BAS files) for your games/programs, which practically eliminates the need to issue commands (you just run the .BAS file), thus, **makes loading process much easier**:



It supports direct file folders, DSK images and SNA snapshot files. It will be especially useful for CPC 464, where giving RSX commands with arguments is rather tiresome (instead of giving directly commands,like with CPC 6128, you need to set first a string variable, for example, the command: |CD,"DIR", is given as: A$="DIR":|CD,@A$).

CPC Loader: Extract files from dsk images

Fast ‘Direct load’ mode, requires to have access, directly to the files of a game/program. Using this utility (**Cpcload folder**, can be found **here**):



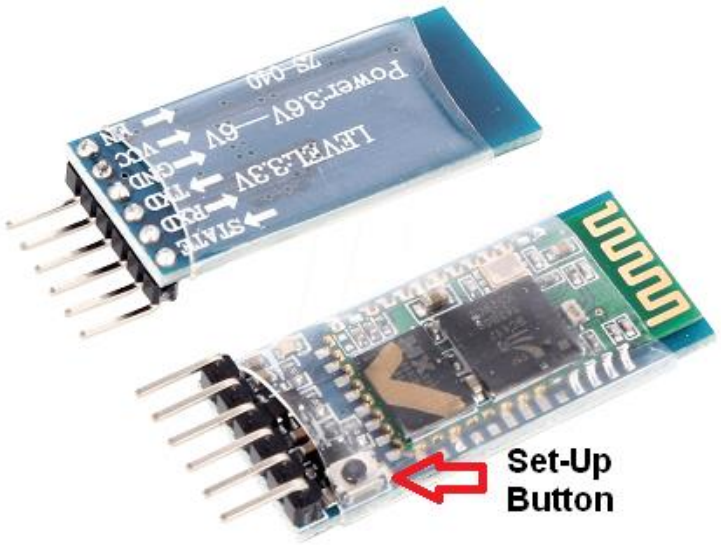
You can easily extract files from multiple dsk images and have them in separate folders, almost instantly! Then, you can also use Amstrad cpc loader creator, to make a Basic loader for each game!

Other

ESP8266_flasher.zip: Contains all required tools and files for easily flash an ESP 8266 WiFi module.
loaders.zip: Ready made list of Basic Loaders for all games in **“Classic & Worth to Play”** list.
Help.txt: All RSX commands in a single page for easy print out! ;-)
Multi_dsk_manager.zip: A Dsk manager for easy mounting images manually, to the 4 available slots.
Inner workings folder: Here you can find all assembly code used in the Usifac II project, along with board schematics, and hex file/assembly/gcb source code for Board’s controller (PIC18F47Q10 MCU).
SNA.zip: A Snapshot file collection of more than 350 games, with ready made Basic loaders for easy loading. Just copy the entire SNA folder to a usb stick and choose any BAS file for direct loading of game snapshot!
setservr.bas: A small utility for easy setup ESP8266 WiFi module as TCP server.
connect_to_bbs.zip: Program+instructions to connect to a bbs, using an esp8266 WiFi module.

Configure Bluetooth Module

The recommended Bluetooth module is HC-05:



You can also use the HC-06 which is a bit cheaper, but it lucks the Set-Up button, which is very useful, if you want to change the default speed of the module (9600bps).
You can easily find these on ebay or electronics shops for a few euros.
Of course you will need a pair of these, to establish a bluetooth connection between Amstrad CPC and a PC or another Amstrad CPC. Or, you can use only one module for Amstrad CPC serial interface to connect with a tablet or a smart phone! ;-)
Connect Bluetooth module to the USB2Serial adapter cable (Red: 5V, Black: Ground, Green: Tx-goes to Rx pin, White: Rx goes to Tx pin). Then, connect cable's usb plug to a free usb port of your PC, **WHILE pressing the setup button**. The red led should flash every 2-3 seconds.Now, you should set serial speed to 38400 by giving |SET command. Then, you can easily configure Bluetooth modules using a terminal program on PC, or the |COM RSX command on Amstrad.
First give the commands:
AT -You should get respond OK
AT+UART=X,0,0
X, can be 2400,9600,19200,38400,57600,115200,230400,460800,921600. You should get response OK.
Now, disconnect the Bluetooth module and connect the other one (don't forget to press the button while powering to enter the setup mode).Give the above commands, but also add this one:
AT+ROLE=1 –You should get response OK
This is for making the module Master. In order for two Bluetooth modules to connect to each other, one must be Master and the other Slave. But Since all BT modules are preconfigured as Slave, you should change one of them to Master.
Finally, power off and on the BT module (in order to exit from setup mode) and connect the other BT module to serial interface. After a few seconds, **both modules should flash twice every 3-4 seconds**. This means that they are connected, and you are ready to go!

Configure ESP8266 WiFi Module

Important note:if you want to **use WIfi module permanently, deactivating "auto usb" function is mandatory** (just give **OUT &FBD1,**93 to do it) , because with "auto usb" enabled, every time you reset Amstrad CPC, it will change the serial port speed (and maybe block windows utility too), and you will need to reconfigure connection with |WIFI.

You can directly connect various versions of ESP8266 WiFi modules, by directly plug them into ULIfAC's ESP 8266 WiFi module 4pin port . The following modules are tested and work ok:

Esp-07

Esp-12F/E

Esp-01/01S

The latter, will also need this 5v adapter (ESP-01 adapter):

Which will be combined to this:

You can easily find any of these, on ebay or many electronics shops, at very cheap prices, of only a few euros. Now, in order to use WiFi, module must be connected to an Access point and then connect to the TCP server that is activated by the PC utility program. This can be done either manually (by giving

the appropriate AT commands), using the small terminal utility through |COM command, or more easily, by giving **|WIFI** command. Program will first check for the current Wifi module serial speed, and ask you if you want to change it.
Then, you just give the name and password of the access point (where WiFi module will be connected to),and then, the ip address and port of the TCP server ,taken from groupbox **WiFi Settings** in PC utility window program.
If everything goes well, you will get a connected! message, and then you can directly run and use any utility you want. If there is a problem connecting to the given Access Point or TCP server, you will be prompted for retrying or give new settings.
If for any reason you want to change TCP server's port number, you can do it from **Wifi Settings** group box, (you will have to close and reopen the PC utility for activating the new port setting) .
Note also that, when a WiFi Connection is established, the **Connection button** will be automatically enabled and connection type will be set to: **Connection: WiFi**. You can also have a serial connection and a WiFi connection both activated at the same time and select which one to use by pressing the connection type button.
In order to disable WiFi module connection you must first send a single string of: +++ (check **AT+CIPMODE** command bellow) and then give the AT command: **AT+CIPCLOSE** to disconnect from TCP server.
ESP8266 module use AT commands for control and communication.
You can give all AT commands using terminal program, or directly, using this small BASIC program (**a$** is the AT command string):

```
10 FOR k=1 TO LEN(a$)
20 OUT &FBD0,ASC(MID$(a$,K,1))
30 NEXT K
40 OUT &FBD0,13: OUT &FBD0,10
```

Note that after giving the AT command, an: **out &fbd0,13: out &fbd0,10** (CR and Line Feed) must be also sent for proper execution!

There is a **very big manual** with 100's of commands available, but bellow, i give you the most useful commands, for configuring the module manually:

AT+CIPMODE= 0 or 1 Sets Transmission Mode
0: normal transmission mode (must be used when WiFi acts as a TCP server)
1: UART-Wi-Fi passthrough mode (transparent transmission), this mode is used for communication with the PC utility program. It can only be enabled in TCP
single connection mode Notes:
• The configuration changes will NOT be saved in flash.
• During the UART-Wi-Fi passthrough transmission, **if the TCP connection breaks, ESP8266 will keep trying to reconnect until +++ is input to exit the transmission**. If it is a normal TCP transmission and the TCP connection breaks, ESP8266 will give a prompt and will not attempt to reconnect.

AT+CIPSEND Start sending data in transparent transmission mode.
Data must be sent with **a 20-ms interval between each packet**, and a **maximum of 2048 bytes per packet.**
To **disable** transparent transmission mode, a **single packet** containing **'+++'** must be received**, (**in practice, you must give: **out &fbd0,43: out &fbd0,43: out &fbd0,43)** ESP8266 returns to normal AT command mode.Wait for at least one second before sending the next AT command.
This command can only be used in transparent transmission mode which requires single connection. Note also that, **you CANT give AT commands when transparent transmission mode is activated.**

AT+CIPSEND=[link id],x send x bytes. Bytes/chars follow immediately after the command. Link id is usually 0 for single connections, but can be up to 5 in a LAN situation

AT+CIPCLOSE Closes the TCP/UDP/SSL Connection.
AT+CWLAP Will list access points in range.
AT+CWJAP=yourSSID,yourWifiPassword connects to an access point
AT+CWQAP quit from acess point
AT+UART_CUR=115200,8,1,0,0 (don't remain after reboot/power off)
AT+UART_DEF=115200,8,1,0,0 Set UART speed (remains after reboot/power off)
AT+CWMODE = 1 (1 for standard mode , 2 for AccessPoint mode & 3 for both)
AT+CIPMUX=1 (set multiple connections, this must be set to '1' if you want to use module as TCP server)
AT+CIPSERVER=1,80 (start the TCP server at port 80)
AT+CIFSR (shows WiFi module's ip/mac address)
ATE0 remove echo (recommended for having WiFi module act as TCP server)
AT+CIPSTO=X (server timeout, range between 0 and 7200, set to large number or 0 for never)
AT+CIPSTART=TCP,Tcp server IP,port no. Connects to a TCP server, useful for connecting many Amstrad CPC's together in a WiFi network!
AT+CWSAP_DEF=<ssid>,<pwd>,<chl>,<ecn> Set up WiFi module's Access point
<ssid>: string parameter, SSID of AP.
<pwd>: string parameter, length of password: 8 ~ 64bytes ASCII.
<chl>: channel ID.
<ecn>: encryption method; WEP is not supported.

- **0: OPEN**
- **2: WPA_PSK**
- **3: WPA2_PSK**
- **4: WPA_WPA2_PSK**

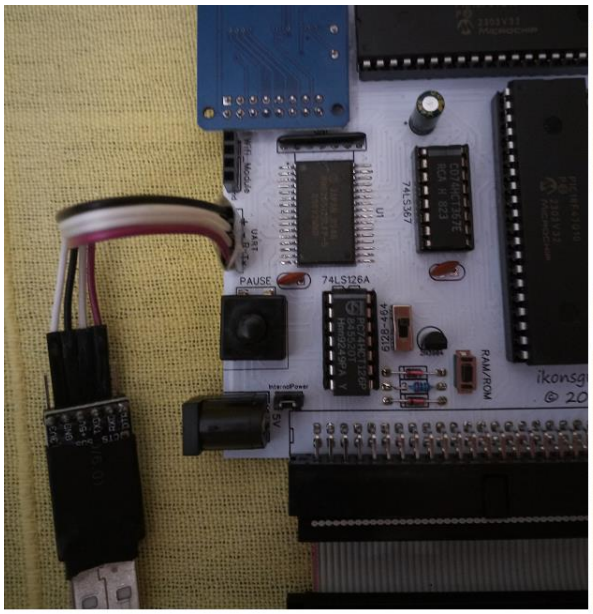
Example: **AT+CWSAP_DEF=ESP8266,1234567890,1,3**

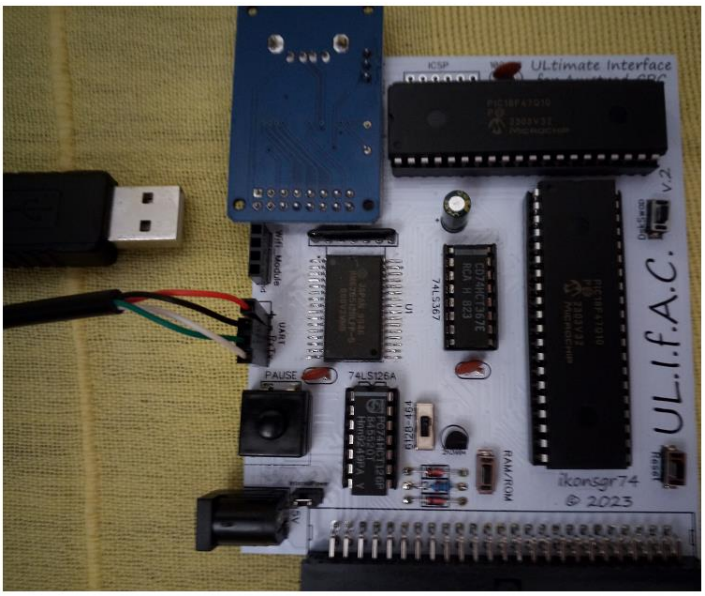
For more detailed information and examples, you can refer to the AT command manual mentioned above.

Updating the Firmware of ULIfAC **(Using on-board UART connector)**

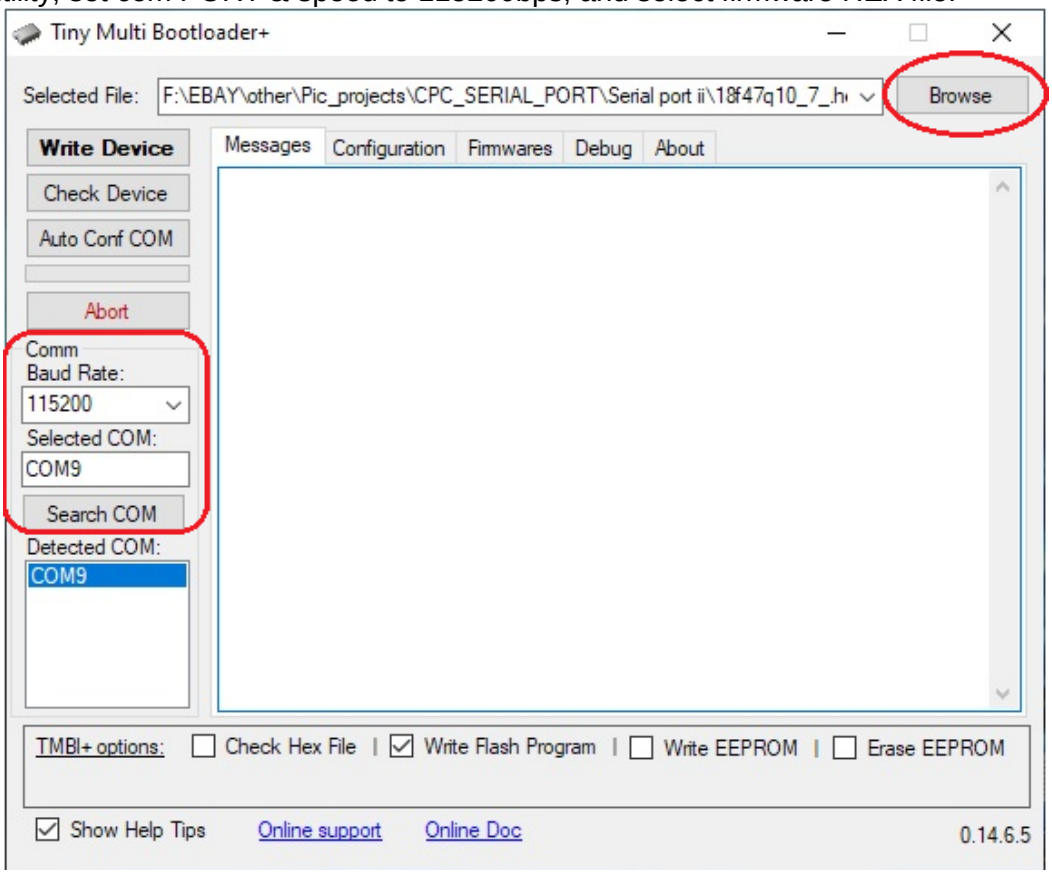
This procedure requires a Bootloader, which is ONLY available with revision 3 (rev3) firmware and newer. You will need :
- usb2serial cable adapter like **this** (PL2303TA) or adapter like **this** (CP2102,for this type you will also need 4 female to female DuPont cables). For the most common PL2303 usb2ttl adapter cables, you can find drivers **here** (also verify which chip version you need for the windows version you use)
- The latest Firmware which can be found **here**

First, Download and install the **TinyMultiBootloaderPlus software** (TBL)
Connect the ULIfAC board to the usb2serial adapter AND THEN to a usb port of your PC:

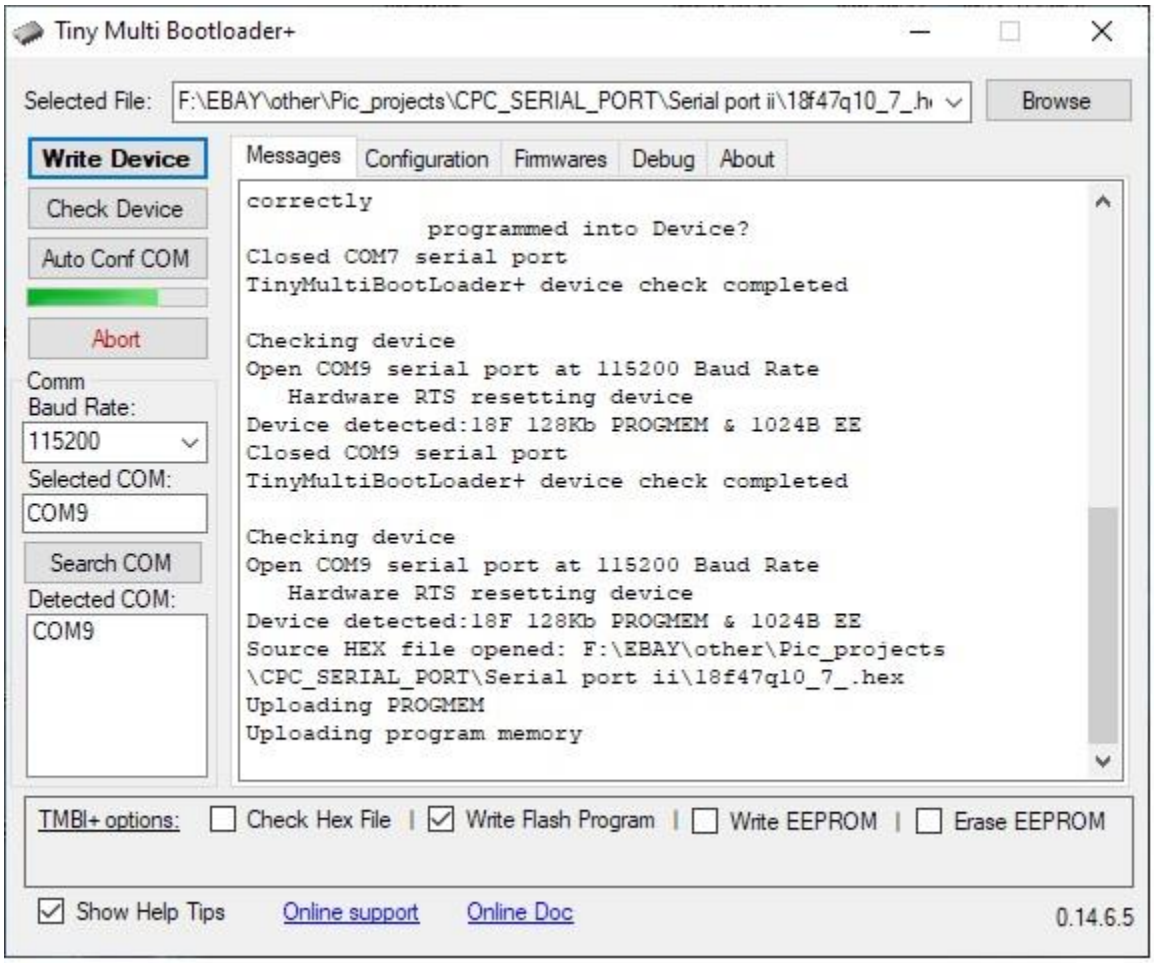




Start TBL utility, set com PORT & speed to 115200bps, and select firmware HEX file:



Click **Write Device** button and immediately the ‘Reset” button on ULIfAC board. Flashing Firmware should start:



After completion of the process (in ~2minutes), disconnect usb2serial adapter , reconnect the ULIfAC card to Amstrad CPC and power up.
At the BASIC prompt enter ‘OUT &FBD2,5’
Restart/Reset the Amstrad CPC and the ULIfAC message should now appear on the Startup Screen
Type ‘|STAT’ to verify all is well. That’s it!

Hints & Tips - Troubleshooting

- **CLEAN THOROUGHLY Amstrad CPC expansion connector before plugging the board!** From my long experience, many times, issues are related to dirt/oxidized edge connector!

- As a general rule, better use **RESET button** (instead of on/off switch) every time you want to restart Amstrad. Also this is the proper action for any problem you might encounter.

- **If you don't get board's boot message "ULIfAC Enhanced" and board doesn't seem to respond,** try giving OUT &FBD2,5, this will reset Board's rom number to default '5' and in most cases, resolve the problem

- **If you boot Amstrad and you get a black screen**,try to press pause button, if it's active, then Amstrad will freeze upon booting!

- If you happen to get an **"unknown command"** error message, when trying to use any of the RSX commands, just give: **OUT &FBD1,4** and everything should work fine. Just remember to give: "**|EN**" again, before load/run/save a game program.

- **If you are using a usb device, and have "Auto USB" feature disable**, remember to give "**|USB**, after initial cold boot, you DON'T need to do it again, no matter how many times you Reset Amstrad.

- To use arguments with RSX commands on a CPC464, you can't pass them directly (this is BASIC's 1.0 "problem"), but you need first to assign argument to a variable. For example, if you want to change directory to "DIR1", you can't just give: **|CD,"DIR1"** but instead:**A$="DIR1" ,**and then:**|CD,@A$**

- **For showing all folders properly**, with <DIR> notification, make sure to **remove the "archive" bit property**, from all files/directories (open usb root directory, select everything ->right click-> select properties->un check 'Archive' check button).

- **Amstrad PLUS is not OFFICIALLY SUPPORTED**, as i don't own a PLUS machine, thus board is never tested on PLUS machines by me. However, there are many other people, who tried the board with **Amstrad PLUS**, and they claim that **for the most part, is functioning ok** , e.g. you will be able to load games from files or dsk images. So any questions/issues regarding PLUS machines they can only be supported by the community.

- **If you want to use WIfi connection permanently, deactivating "auto usb" is mandatory**, because with "auto usb" enabled, every time you reset Amstrad it will change the serial port speed (and maybe block windows utility too) and you will need to reconfigure connection with |WIFI.

- **If windows utility program stops responding,** just click "Connected" button, in most cases this will fix the problem.

- **For any unsolved technical problems, bugs etc,** you can get support by me and the large CPC Wiki community, **here**

Finally, here are a few I/O commands that might be useful if you want to develop your own programs utilizing ULIfAC:

out &fbdd1,4 : Disables direct mode (e.g. all cat/load/save/run commands will be directed to disk drive)
out &fbdd1,5 : Enables direct mode (if usb mode=off then all cat/load/save/run commands will be directed to serial port->connected pc, otherwise, usb flash will be accessed)
out &fbd1,51 :enable/disables usb mode (on->off or off->on)
out &fbd1,8 :disables usb mode
out &fbd1,30 : Get status report (like with |STAT command)
out &fbd1,31 : Get image slots report (like with |SMG command)
out &fbd1,70-73 : Select directly dsk image slot 1-4
inp(&fbd5) : Get currently selected dsk image slot
inp(&fbdd) : Get serial speed code
inp(&fbde) : Get usb mode state (1=enable, 0=disable)
inp(&fbdf) : Get FDC emulation state (1=enable, 0=disable)