AC – DC echipamente electronice srl
Cluj-Napoca, str. Muncitorilor, nr. 18, ap 8
tel: 0745-659909 www.acdcelectronics.ro

USB-UART 3.3V converter for WiFi ESP-01 module

The converter with FT232 is especially designed for the ESP-01 module.
The usage is comfortable for:

- Flashing, running and debugging the programms.
- Sending AT commands.
- Uploading a new firmware.

The convertor is achieving:

- The control of Reset and GPIO0 pins with the help of the built in switches.
- The convertion between USB COM port and UART 232 (Tx, Rx), logical level 0~3,3V.
- Led visualisation of Tx and Rx signals.
- Power supplying ESP-01 module with 3,3V, max. current 0,5A.

Fig. 1

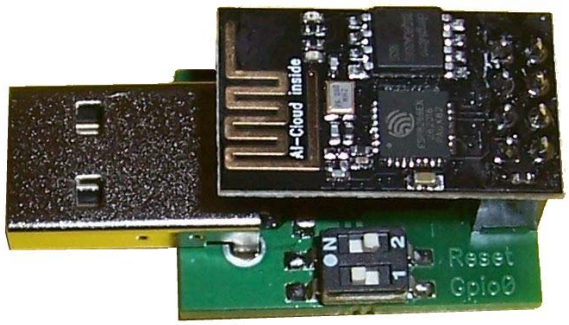


Fig. 2

ESP-01 was released in august 2014 and is built around the ESP8266 circuit made by a Chinese company. The low price, high performances and the open source promotion have captured the interest in the WiFi domain.

In a short time, the world community has expanded the programming of the ESP-01 module in various IDE: Arduino, Python, Lua.

Note 1: This tutorial is useful to anyone working for the first time with ESP-01.
The presented examples are made in the **Arduino** IDE.
The described procedures are similar in any other IDE.

Loading the ESP8266 libraries into the Arduino IDE

Open this reference link. **Ignore** the hardware steps and make only the software stage 2.

The files are loaded in:
C:\Users\<user>\AppData\Local\Arduino15\packages\esp8266\hardware\esp8266WiFi

In this way, the ESP8266 circuit (ESP-01, ESP-12 etc) becomes a new Arduino shield.
The communication is serial UART 232 (Tx, Rx), logical level 3,3V.

Uploading the Arduino programm in ESP-01

Continue with step 3 in the reference link. You don't need the hardware components as required in the reference link e.g. Arduino board, buttons, cables etc.

Just use our converter (fig.2) connected to an USB port of your PC.
Select this port in Arduino/Tools/Port. Select Upload Speed 115200.

Select Generic ESP8266 Module in Tools/Boards.

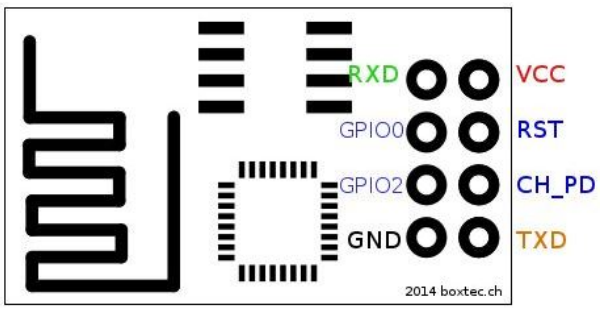
Open the Led Blink programm in Examples/01.Basic/Blink.

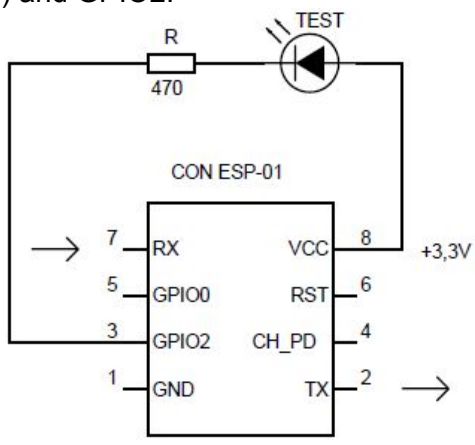
Take into account, ESP-01 has only two digital I/O pins: GPIO0 and GPIO2.
Change the output pin value to 2.

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 2 as an output.
  pinMode(2, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(2, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(2, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

Solder a led and a 470Ω~1k resistor between Vcc (+3,3V) and GPIO2.





Note 2: It's preferable the load (led and resistor) to be soldered between Vcc and GPIO2 (or GPIO0). In this way, the load does not disturb the control signal GPIO0.

Flash the programm into the ESP-01 module. There are three steps:

1. Put the GPIO0 switch in ON (to GND). In this way ESP-01 is ready for programm loading.
2. Reset the ESP-01 module. For this purpose put the Reset switch in ON (to GND) and than return to OFF state.
3. Select Upload in Arduino/Sketch/Upload.

During the programm loading, Arduino shows:

"Sketch uses 222,201 bytes (51%) of program storage space. Maximum is 434,160 bytes.
Global variables use 31,576 bytes (38%) of dynamic memory, leaving 50,344 bytes for local variables. Maximum is 81,920 bytes.
Uploading 226352 bytes from C:\Users\Emil\AppData\Local\Temp\build89dffa342730d36061756adb8c22691b.tmp/My_Blink_ESP.ino.bin to flash at 0x00000000
..... [36%]
..... [72%]
..... [100%]"

After loading, the ESP-01 module passes immediately to the programm running. The led is lighting with 2 seconds period.

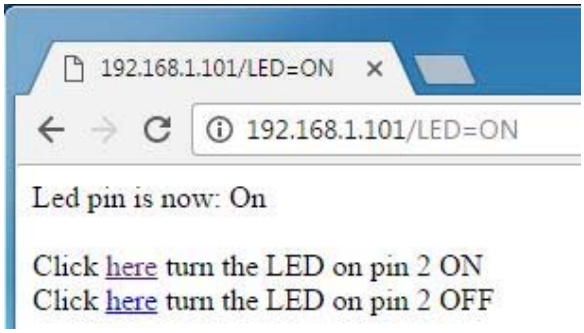
At a new power up, the programm runs only if GPIO0 and Reset are OFF. The reset function is automaticaly done.

Note 3: Together with the first Arduino programm flashing it is erased any previous firmware, in this case the factory firmware (AT + Commands interpreter).

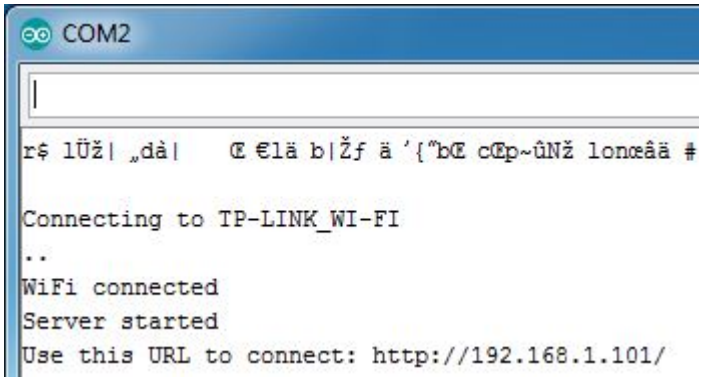
A very useful programm is esp-01_web_1_led.ino and is using the already mounted led.

You'll command this led in any browser with the ESP-01 own loaded web page.

You can use a PC or a smart phone connected to the WiFi router.



The ESP-01's wireless LAN address where this web page exists is shown immediately after reset in Arduino serial monitor or another terminal:



The same programm, modified to switch two leds is esp-01_web_2_leds.ino.

Note 4: Exactly identify your router's ssid and password. These values can be found on the back router's label, but not always the ssid writen on the label corresponds to the visible ssid in your router's page, e.g. 192.168.1.1.

In our case, the writen ssid on the router's label is TP-LINK_15CA, but the autoconfigured ssid in the router's page is TP-LINK_WI-FI.

You can trust the label's printed password (8 digits), as long you have not modified it previously, using the administrator's privileges.

Note 5: In this way you can upload in ESP-01 and check other examples included in:

C:\Users\<user>\AppData\Local\Arduino15\packages\esp8266\hardware\esp8266WiFi\src\ESP8266WiFi.h\examples

Note 6: A program to reduce the current to 0.5mA is esp-01_client_light_sleep.ino.

These examples help you to design your own desired programms.

Sending AT commands into a new ESP-01 module

According to this material, the factory firmware is AT + Commands interpreter.

The module answers to this AT commands set, sent manually or by another controller.

Before the first programm loading, it's helpful to manually send some AT commands.
On this occasion you'll get the baud rate value confirmation, usualy 115200.
Only the first module series used 9600 baud.

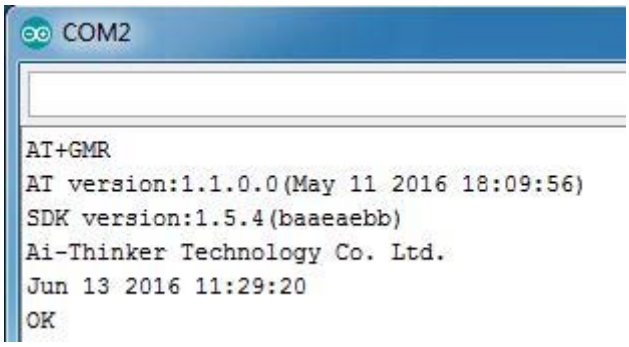
Use our converter (fig.2).

Put the Reset and GPIO0 switches in OFF state.

Open any terminal e.g. Serial Monitor Arduino. In the box No line ending select Both NL & CR.

In this way, the sent AT commands are finalised with CR si LF.

Send AT and AT+GMR commands succesively:



The answer should be OK.
In this case the ESP-01 module is new or at least was not previously loaded with any programm.

Note 7: Once the first program has been loaded, the AT commands sending is not possible.
Don't think an ESP-01 module is damaged when not answering to AT commands!
It means the module has already a programm loaded, good or bad.
Just reload the factory firmware, as shown in the next chapter.

Uploading a new firmware in ESP-01

We have shown, together with an Arduino programm loading, any previous firmware is erased.

The procedure for reloading the factory firmware (AT + Commands interpreter) or another personalised firmware e.g. NodeMCU Lua or Python is presented in this link.

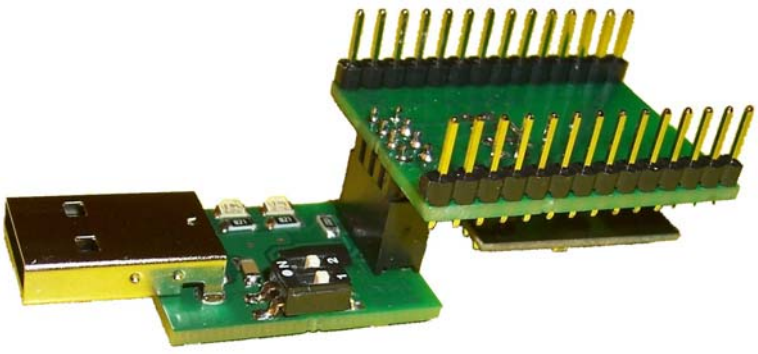
You need:

- The firmare files with .bin extension.
- The software tool for uploading firmware files.

Prepare the ESP-01 module for uploading files as shown in the previous chapter.

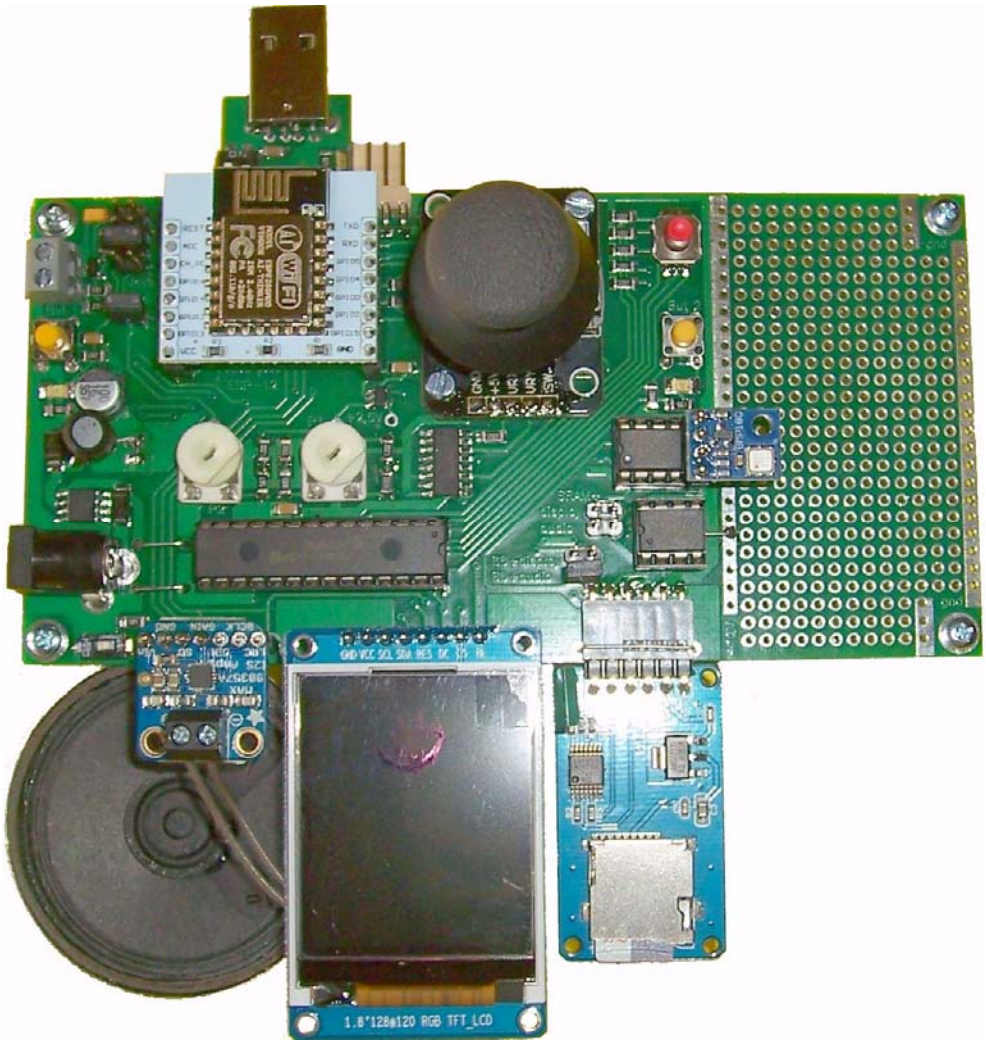
Uploading the programm in the WiFi adapter for Mini PLC4404

WiFi adapter is especially designed for Mini PLC4404.



The convertor use in the WiFi ESP-12 module case

The ESP-12 test board is designed to study and experiment the ESP8266 circuit in various applications.



The test board is using the ADC and GPIO pins and helps to experimentaly get the right ADC analog reference Aref value.

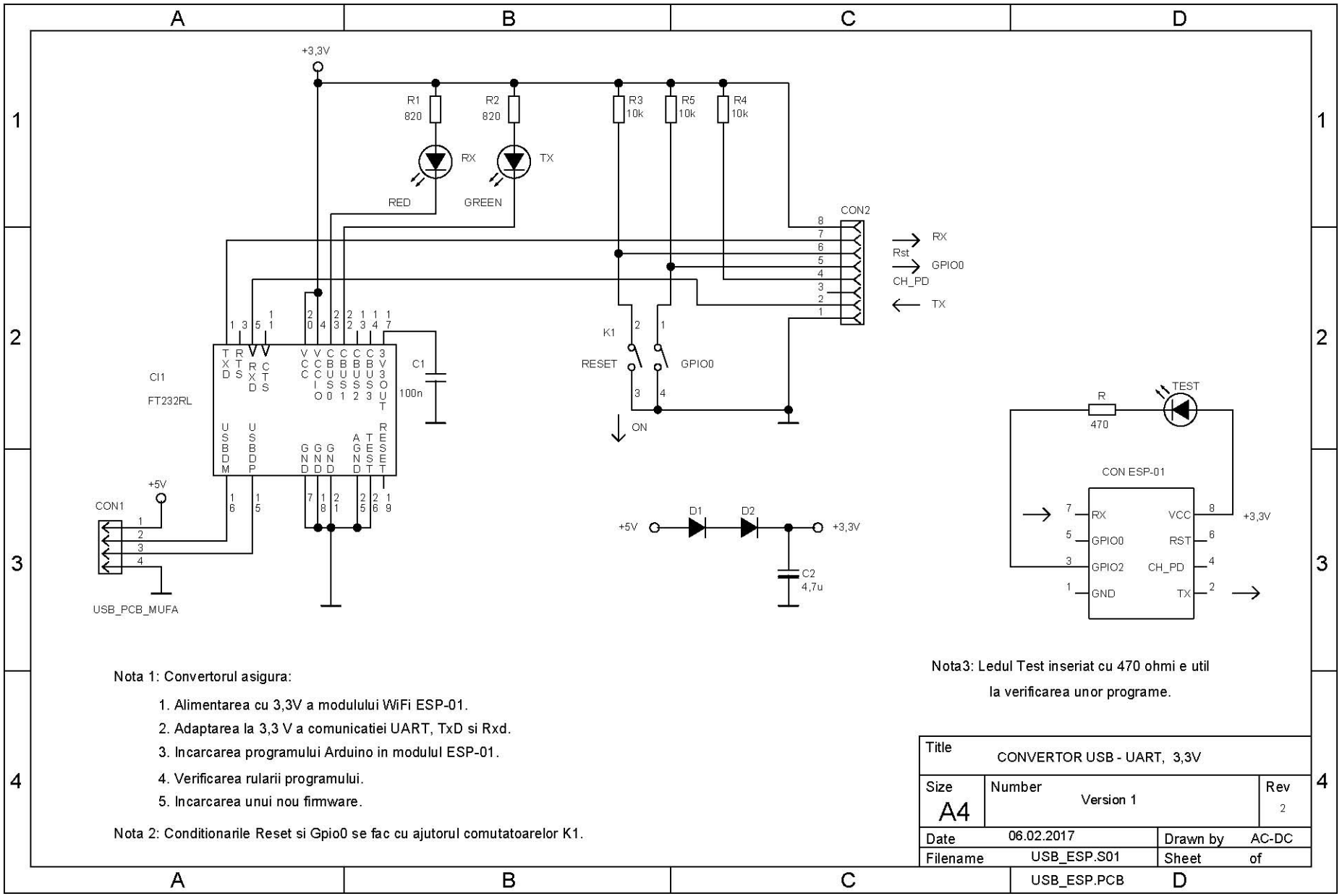
The programm loading is done using our converter, or another method.

According to development board's electronics sketch, there are two ways to run the programms:

- +3,3V supply and serial messages visualisation using our converter
- 5~12V external supply and serial messages visualisation using our reader USB device of the UART serial traffic

The electronics sketch of the USB-UART 3,3V converter

This is our free contribution to ESP-01 open source promotion.


A
B
C
D
1
2
3
4
+3,3V
R1
820
R2
820
R3
10k
R5
10k
R4
10k
RX
TX
RED
GREEN
CON2
RX
Rst
GPIO0
CH_PD
TX
K1
RESET
GPIO0
ON
CI1
FT232RL
C1
100n
CON1
+5V
USB_PCB_MUFA
D1
D2
+5V
+3,3V
C2
4,7u
R
470
TEST
CON ESP-01
RX
GPIO0
GPIO2
GND
VCC
RST
CH_PD
TX
+3,3V
Nota 1: Convertorul asigura:
1. Alimentarea cu 3,3V a modulului WiFi ESP-01.
2. Adaptarea la 3,3 V a comunicatiei UART, TxD si Rxd.
3. Incarcarea programului Arduino in modulul ESP-01.
4. Verificarea rularii programului.
5. Incarcarea unui nou firmware.
Nota 2: Conditionarile Reset si Gpio0 se fac cu ajutorul comutatoarelor K1.
Nota3: Ledul Test inseriat cu 470 ohmi e util
la verificarea unor programe.
Title
CONVERTOR USB - UART, 3,3V
Size
A4
Number
Version 1
Rev
2
Date
06.02.2017
Drawn by
AC-DC
Filename
USB_ESP.S01
Sheet
of
USB_ESP.PCB