

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs

CC BY-NC-ND



Codul sursa din acest document este licentiat

Public-Domain

Esti liber sa distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, printare, sau orice alt mijloc), atat timp cat nu aduci nici un fel de modificari acestuia. Codul sursa din acest document poate fi utilizat in orice fel de scop, de natura comerciala sau nu, fara nici un fel de limitari.

# WiFi Car Reloaded (partea a II-a)

Adăugarea unei camere video la o platformă mobilă (mașină teleghidată) oferă posibilitatea extinderii funcționalității exploratorii – utilizatorul va putea să controleze la distanță platforma mobilă văzând efectiv mediul în care aceasta se mișcă. Această facilitate este folosită uzual la roboții teleghidați ce explorează medii periculoase (roboți de deminare de exemplu) sau medii inaccesibile omului. WiFi Car Reloaded își propune adăugarea acestei funcționalități pentru a crește cazurile de utilizare posibile.



Pentru a nu crește costul sistemului vom utiliza o cameră web USB obișnuită dar se poate modifica foarte ușor proiectul pentru a utiliza o cameră serială de exemplu. Calitatea camerei web va da performanțele de achiziție a imaginilor pentru sistem. Conectarea unei camere web USB la un sistem ce rulează OpenWRT a fost explicată în lecția "Ce putem face cu o cameră web veche?" dar vom relua pe scurt elementele principale ale configurației în cele ce urmează.



Conectarea unei camere web USB la placa LinkIt Smart 7688 Duo necesită un adaptor OTG (imagine alăturată). După conectarea fizică a camerei trebuie să verificăm recunoașterea corectă a dispozitivului USB de către sistemul de operare, se face cu ajutorul următoarelor comenzi (în consola OpenWRT):

`dmesg` – va afișa mesajele generate de nucleu sistemului de operare la conectarea dispozitivului USB;


```
[ 244.840000] usb 1-1: new high-speed USB device number 2 using ehci-platform
[ 245.040000] usb 1-1: no of_node: not parsing pinctrl DT
[ 245.040000] uvcvideo 1-1:1.0: no of_node: not parsing pinctrl DT
[ 245.040000] uvcvideo: Found UVC 1.00 device USB2.0 PC CAMERA (1908:2311)
[ 245.050000] input: USB2.0 PC CAMERA as /devices/101c0000.ehci/usb1/1-1/1-1:1.0/input/input0
root@mylinkit:~#
```

`lsusb` – va enumera dispozitivele USB conectate;

```

root@mylinkit:~# lsusb
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 001 Device 002: ID 1908:2311 GEMBIRD
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
root@mylinkit:~#

```



`lsusb -s 001:002 -v | egrep "Width|Height"` - ne permite identificarea modurilor video – rezoluțiilor – suportate;

```

root@mylinkit:~# lsusb -s 001:002 -v | egrep "Width|Height"
Width          640
Height         480
Width          352
Height         288
Width          320
Height         240
Width          176
Height         144
Width          160
Height         120
Width( 0)      640
Height( 0)     480
root@mylinkit:~#

```

`ls /dev/video*` - va enumera dispozitivele video din sistem.

```

root@mylinkit:~# ls /dev/video*
/dev/video0
root@mylinkit:~#

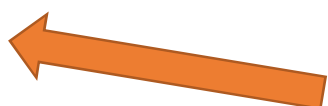
```

În cazul în care una dintre comenzile prezentate nu este recunoscută sau camera video nu este recunoscută ca dispozitiv USB sau video se va verifica dacă pachetele software necesare sunt instalate:

```

opkg update
opkg install kmod-video-uvc kmod-video-core
opkg install usbutils
opkg install mjpg-streamer

```



Pentru transmisia live de imagine se va utiliza programul **mjpg-streamer** . Comanda de pornire a acestuia este:

```

mjpg_streamer -i "input_uvc.so -d /dev/video0 -y" -o
"output_http.so"

```

Accesarea imaginilor transmise se poate face la adresele:

Flux video:

[http://adresa\\_IP\\_sistem:8080/?action=stream](http://adresa_IP_sistem:8080/?action=stream)

Imagini:

[http://adresa\\_IP\\_sistem:8080/?action=snapshot](http://adresa_IP_sistem:8080/?action=snapshot)

Pentru ca aplicația mjpg-streamer să pornerască în mod automat trebuie ca în fișierul */etc/config/mjpg\_streamer* să configurați parametrul enabled (și alți parametrii dacă este cazul):

```
config mjpg-streamer 'core'
    option enabled '1'
    option input 'uvc'
    option output 'http'
    option device '/dev/video0'
#    option resolution '320x240'
    option yuv '1'
    option quality '80'
    option fps '15'
#    option led 'auto'
#    option www '/www/webcam'
    option port '8080'
#    option username 'openwrt'
#    option password 'openwrt'
```

și să executați comanda:

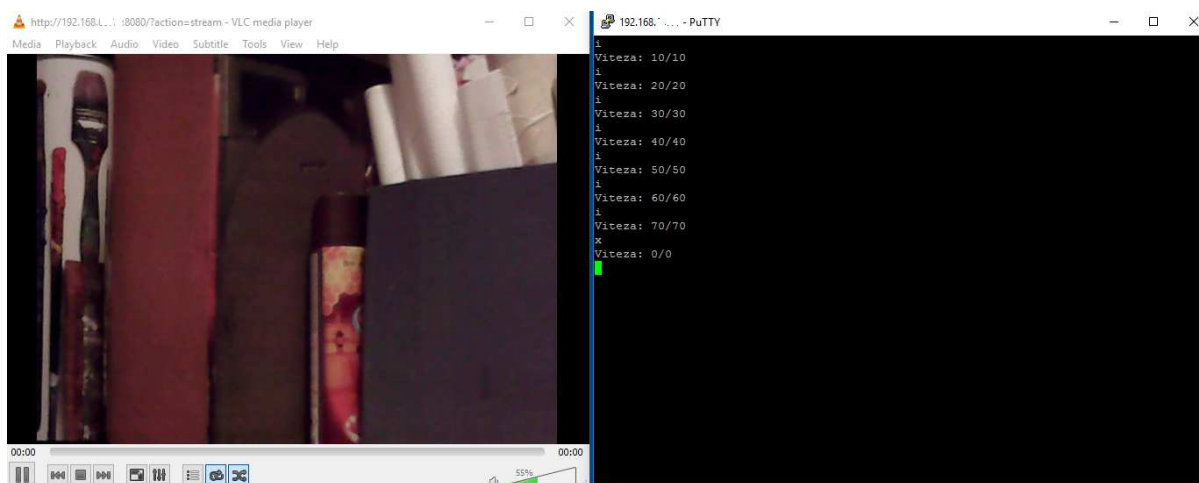
```
/etc/init.d/mjpg-streamer enable
```

Pentru mai multe informații despre configurarea programului mjpg-streamer puteți consulta materialul:

Webcam with the Linux UVC driver [OpenWrt Wiki]

<https://wiki.openwrt.org/doc/howto/webcam>

În acest moment mașina emite imagini live și poate fi comandată simultan prin WiFi. Pentru vizualizarea imaginilor se poate utiliza un client web (browser) sau o aplicație multimedia (VLC de exemplu – captura de ecran de mai jos).



Pentru a putea realiza o comandă mai bună a platformei robotice vom înlocui clientul telnet utilizat până acum cu o aplicație mobilă pentru dispozitivele Android: RoboRemoFree - Arduino control:



<https://play.google.com/store/apps/details?id=com.hardcodedjoy.roboremo>

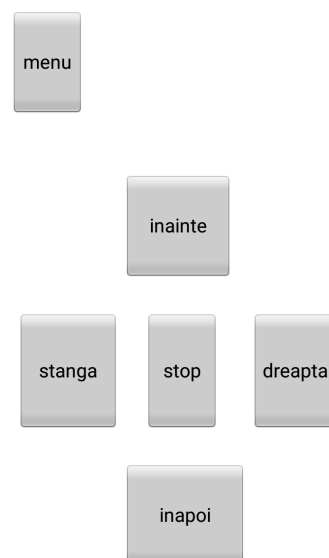
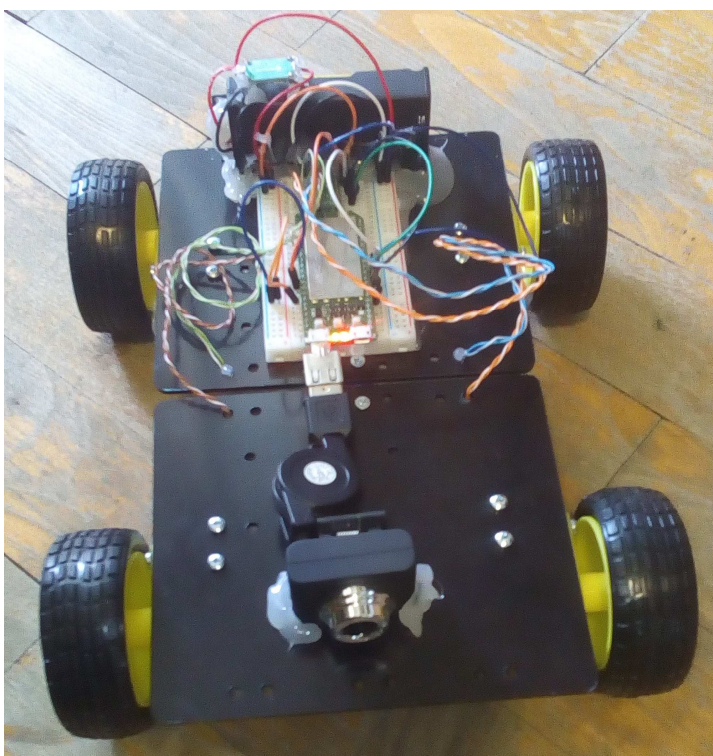
Această aplicație a fost creată pentru proiecte bazate pe platforma Arduino și module de comunicație bluetooth sau pentru platforme bazate pe circuitul WiFi ESP8266 dar noi o vom putea utiliza fără probleme pentru platforma noastră. Mai multe despre proiectul Robo Remo puteți găsi la adresa:

RoboRemo - Customizable Bluetooth / Internet / WiFi Remote Control App

<http://www.roboremo.com/>

Pentru configurarea aplicației mobile trebuie să edităm interfața utilizator (edit ui) și să adăugăm cinci controale de tip buton pe care definim acțiunile (set press action): i, d, s, b, x. După finalizarea interfeței ne vom conecta prin rețea (Internet (TCP)) la adresa *adresa\_ip\_sistem:2001*. În acest moment mașina poate fi comandată prin intermediul aplicației mobile iar timpul de reacție este mult mai bun față de varianta prin intermediul protocolului HTTP.

<https://www.robofun.ro/forum/>



Un ultim sfat: pentru a scădea încărcarea sistemului OpenWRT puteți activa doar comunicația între portul *2001* și a portului serial */dev/ttyS0* . În fișierul */etc/ser2net.conf* se vor comenta toate liniile de după linia ce definește comunicația utilizată:

```
2001:raw:600:/dev/ttyS0:9600 NONE 1STOPBIT 8DATABITS XONXOFF LOCAL -RTSCTS
#2002:raw:600:/dev/ttyS1:9600 NONE 1STOPBIT 8DATABITS XONXOFF LOCAL -RTSCTS
#2003:raw:5:/dev/ttyS2:9600
#2004:raw:5:/dev/ttyS3:115200
...
```

Dacă sunteți în căutare de idei suplimentare vă recomandăm și următoarele proiecte:

Wi-Fi RC Car - Qi Enabled - Arduino Project Hub

<https://create.arduino.cc/projecthub/ioanniskydonis/wi-fi-rc-car-qi-enabled-b972b2>

WiFi Controlled Mobile Robot | Adafruit Learning System

<https://learn.adafruit.com/wifi-controlled-mobile-robot/>

Arduino RC Car With FPV Camera

<http://www.instructables.com/id/Arduino-RC-Car-with-FPV-Camera/>

Make a remote tank with LinkIt smart 7688 Duo and MCS gamepad channel

<https://iamblue.gitbooks.io/linkit-smart-nodejs/content/en/cloud/gamepad.html>

arduino WiFi remote control car – WiFi arduiCar

<http://wiznetmuseum.com/portfolio-items/arduino-wifi-remote-control-car-wifi-arducar/>

Lukse.lt » Carambola powered robot v2

<http://lukse.lt/uzrasai/2013-01-carambola-powered-robot-v2/>