iocafe

open source IoT/IO device communication

home get involved services docs download blog

Democar project

Overall description

Democar project demonstrates use case of Open Source iocom library in IoT device development. Code of the project is in github Open Source repository with MIT license.

Started with regular RC car

Server is from iocafe repository and it is running on Amazon AWS cloud. Car has ESP32 micro-controller connected to internet with built in WiFi in secure way. User interface is I-spy, self configured python program included in iocafe repository.

Program development is done using provided Linux virtual machine.

posts

- Installing NASA Core Flight System and OpenSatKit to Virtual Machine
- PlatformIO is great
- Democar project
- ESP32: 1 MHz clock signal to output pin
- Configuring micro-controller WiFi with Android phone's LED flash
- IOCOM in a simple multiplayer game?
- iocom secure network topology
- testing development tools and secure communications

comments

Marco Roda on Configuring micro-controller WiFi with Android phone's LED flash

Get Your Microcontroller Online at the Speed of Light – UpMyTech on

ESP32 operating voltage is taken from 4xAA battery pack. Car has been originally RC controlled. Antenna is removed and controlling outputs from micro-controller are routed to electronics board in the car and they override original radio control control signals.

Program development and interfacing microcontroller to car electronics is very fast and easy.

Security is built in and communication is very fast with iocom and car response is immediate to control inputs.

Connecting ESP32 to car electronics

Opening the car was easy, because it is constructed with screws. Everything opens up and control board is very accessible and can be easily modified using cutter and some soldering and using breadboard to accommodate ESP32.

There is one IC in circuit board. It is radio control receiver and by radio control is possible to give essentially four commands: Left, Right, Forward, Backward. Steering is controlled with electric motor and it has spring loaded centering and adjustment of straight line zero position when steering control is centered.

Drive motor has only two controls, full speed forward and full speed backward. Car is very fast and agile and therefore good for demonstrations of fast speed of IoT control over internet using iocom library.

Control board is modified so, that four resistors are desoldered from IC pins and ESP32 digital outputs are connected to resistors overriding IC completely.

Car electronics is 5V system and ESP32 is 3.3V. Care has to be taken that 5V is not connected at any time to ESP32 pins. Also VIN must not be connected to car 4xAA battery when USB is connected to computer.

Configuring micro-controller WiFi with Android phone's LED flash

pekkalehtikoski on Configuring micro-controller WiFi with Android phone's LED flash

pekkalehtikoski on Configuring micro-controller WiFi with Android phone's LED flash

Get Your Microcontroller Online at the Speed of Light | 3d print ......errori ed esperienze, on Configuring micro-controller WiFi with Android phone's LED flash

archived

July 2020

May 2020

April 2020

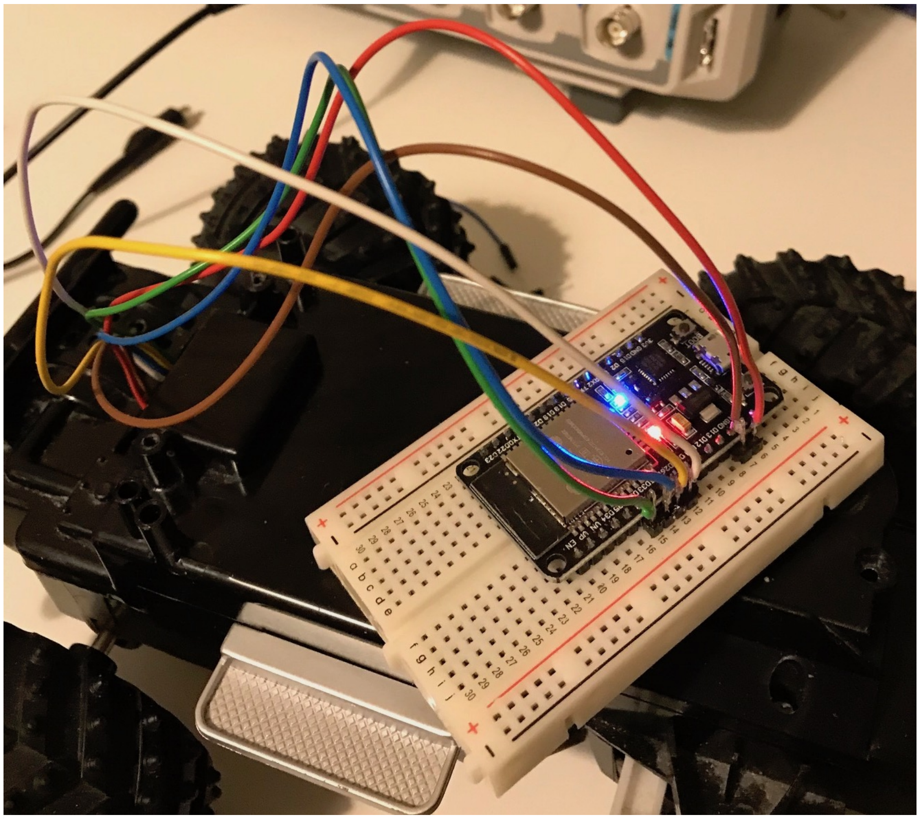
March 2020

December 2019

September 2019

tags

cFS core Flight System Cosmos cubesat democar esp32 espprog io-domain iocafe iocom mac NASA OBC OpenSatKit platformio programming flash release 2020 satellite secure network topology tls virtualbox virtual machine visual studio code



ESP32 connected to democar

ESP32 connections

- RIGHT IC pin 6 resistor 27 GPIO 32 green
- LEFT IC pin 7 resistor 24 GPIO 33 blue
- FORWARD IC pin 11 resistor 17 GPIO 25 yellow
- BACKWARD IC pin 10 resistor 22 GPIO 26 white
- GND brown
- VIN 6V red

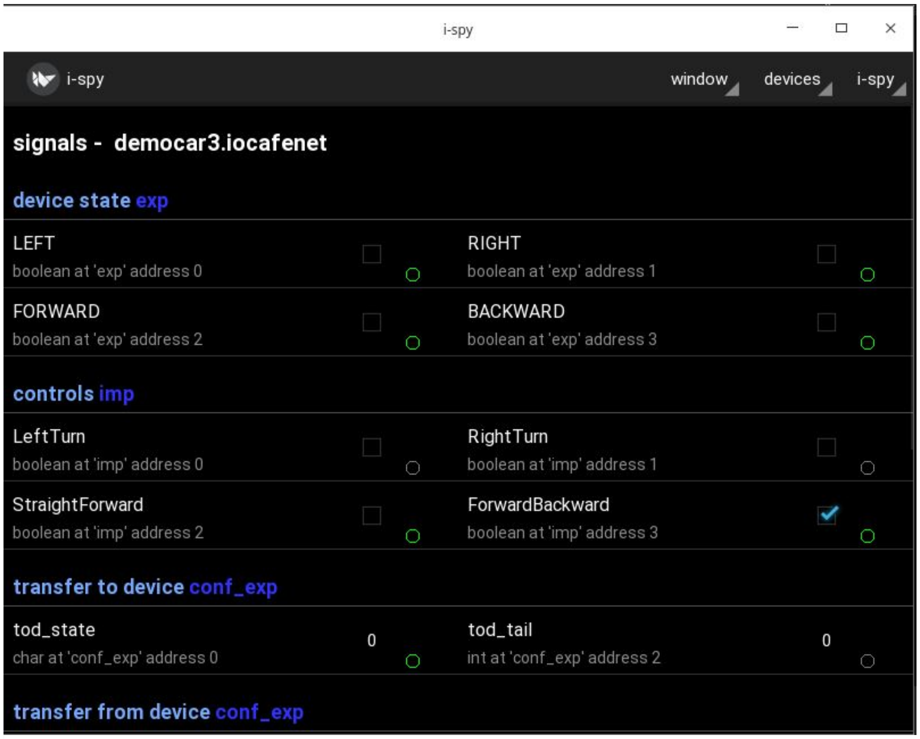
Program development

In Amazon AWS server program ”Frank” is running in ubuntu server. Basic structure of code running in ESP32 is from ”Gina”. It has basically two functions as main structure, initialization function and looping function.

Libraries iocom, eosal and pins are used in the program. Configurations are done using JSON files for network, wifi, pins and signals.

Program is compiled using QT creator and Visual Studio Code. Platform IO is utilized in the project.

User interface test program used in program development, I-spy, provided in iocafe and made with python. It configures automatically from JSON. Also actual pin outputs are shown in user interface. Control inputs are Forward/Backward selection and Movement selection and Movement and Direction selections.



signals - democar3.iocafenet

device state exp

LEFT boolean at 'exp' address 0	☐	○	RIGHT boolean at 'exp' address 1	☐	○
FORWARD boolean at 'exp' address 2	☐	○	BACKWARD boolean at 'exp' address 3	☐	○

controls imp

LeftTurn boolean at 'imp' address 0	☐	○	RightTurn boolean at 'imp' address 1	☐	○
StraightForward boolean at 'imp' address 2	☐	○	ForwardBackward boolean at 'imp' address 3	☑	○

transfer to device conf_exp

tod_state char at 'conf_exp' address 0	0	○	tod_tail int at 'conf_exp' address 2	0	○

transfer from device conf_exp

I-spy user interface with democar controls and signals

Security

Iocom library uses TLS encrypted communication. Server certificate is used to identify server and automatically created username/password is used to identify democar. All devices connected to server are identified explicitly.

Democar in operation

Democar in operation without chassis

Tagged on: democar esp32 iocafe iocom

Markku Nissinen April 18, 2020

iocom library development and testing No Comments

← ESP32: 1 MHz clock signal to output pin

PlatformIO is great →

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

☐

Save my name, email, and website in this browser for the next time I comment.

Post Comment