Navigation and Communication for UGV/UAV

Sachinkumar Omprakash Dubey

Indian Institute of Technology, Hyderabad.

June 18, 2022

Content

Introduction

- This thesis titled "Navigation and communication for UGV/UAV, consists of two parts:
  - The navigation part of this thesis includes utilization of various controllers to implement applications on UGV/UAV hardware.
  - The communication part explores the use of SATCOM and NBIoT for communicating with UAV/UGV.
- UGV and UAV kits are ideal low-cost prototype systems for testing software before scaling it up and putting it on a real ground vehicle or a more complicated UAV system.

UGV kit hardware

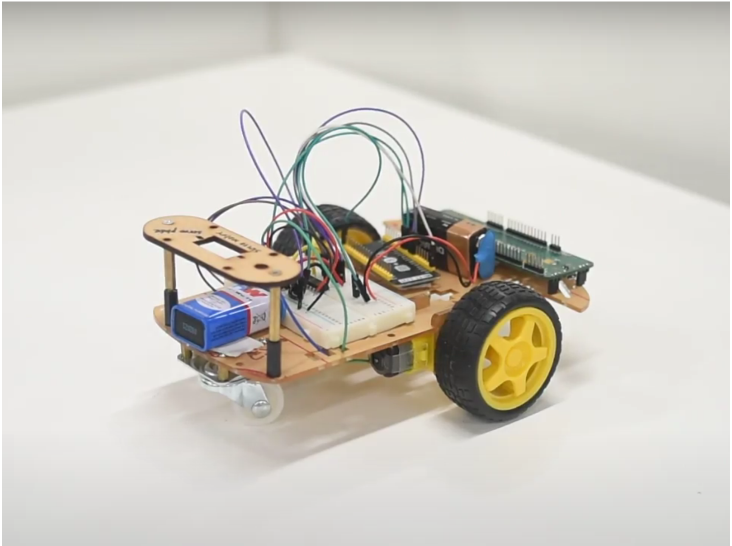


Figure 1: UGV kit hardware

UAV kit hardware

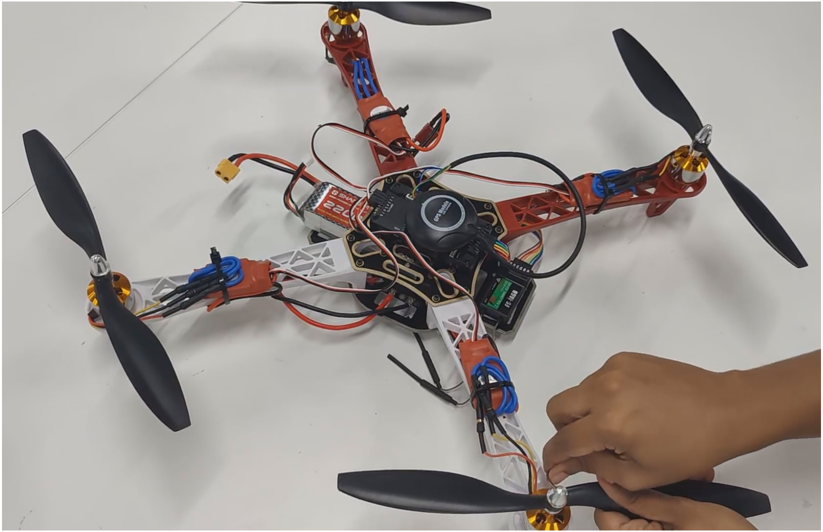


Figure 2: UAV kit hardware

Controllers

Parameters	Arduino Uno	Raspberry Pi 3B	ESP-32
Processor	ATMega328P	Quad-core Broadcom BCM2837 ($4\times$Cortex-A53)	Xtensa Dual-Core 32-bit LX6 with 600 DMIPS
GPU	-	Broadcom VideoCore IV @ 250 MHz	-
Operating voltage	5V	5V	3.3V
Clock speed	16 MHz	1.2GHz	26 MHz – 52 MHz
System memory	2kB	1 GB	$<$45kB
Flash memory	32 kB	-	up to 128MB
EEPROM	1 kB	-	-
Communication supported	IEEE 802.11 b/g/n Bluetooth via Shield	IEEE 802.11 b/g/n Bluetooth, Ethernet Serial	IEEE 802.11 b/g/n
Development environments	Arduino IDE	Any linux compatible IDE	Arduino IDE, Lua Loader
Programming language	Embedded C, C++	Python, C, C++, Java, Scratch, Ruby	Embedded C, C++
I/O Connectivity	SPI I2C UART GPIO	SPI DSI UART SDIOCSI GPIO	UART, GPIO

Table 1: Comparison between Arduino Uno, Raspberry Pi 3B and ESP-32

Controllers (Vaman)

- On-board dual processor (ARM + FPGA)
- On-board WiFi/BT/BLE connectivity with ESP32
- μSD card support
- On-board inertial measurement unit
- On-board BMO055 smart fusion sensor
- On-board DPS310 provides pressure, humidity and temperature monitoring

Figure 3: Vaman - Pygmy BB4

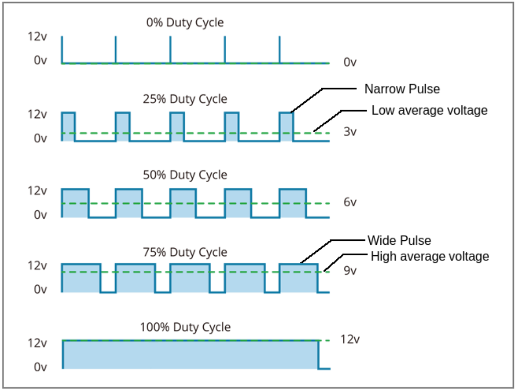
Motor control using PWM

- A pulse width modulation speed control system works by sending a series of "ON-OFF" pulses to the motor. The frequency of square wave is kept constant while varying the duty cycle (the fraction of time that the output voltage is "ON" compared to when it is "OFF").
- By changing the width of the ON duration, one can control the average DC voltage applied to the motor. The below equation (1) gives the relation between the Duty cycle (D) and the average voltage:

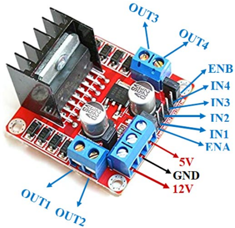
$$V_{dc} = \frac{1}{T}\int_0^T v_{PWM}(t)\,dt \tag{1}$$

$$\begin{aligned} V_{dc} &= \frac{1}{T}\left(\int_0^{DT} v_{\max}\,dt + \int_{DT}^{T} v_{\min}\,dt\right) \\ &= \frac{1}{T}\left(D \cdot T \cdot v_{\max} + T\,(1-D)\,v_{\min}\right) \\ &= D \cdot v_{\max} + (1-D)\,v_{\min} \end{aligned}$$

Motor control using PWM (Continued)



(a) PWM speed control



(b) Dual motor driver module (L298N)

ESP32 Based Application-1

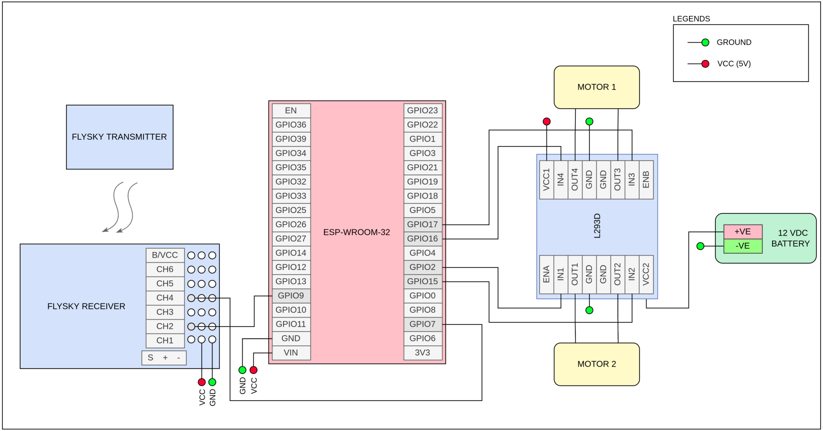


Figure 5: UGV Navigation using Fly-sky transmitter & receiver (ESP32)

ESP32 Based Application-2

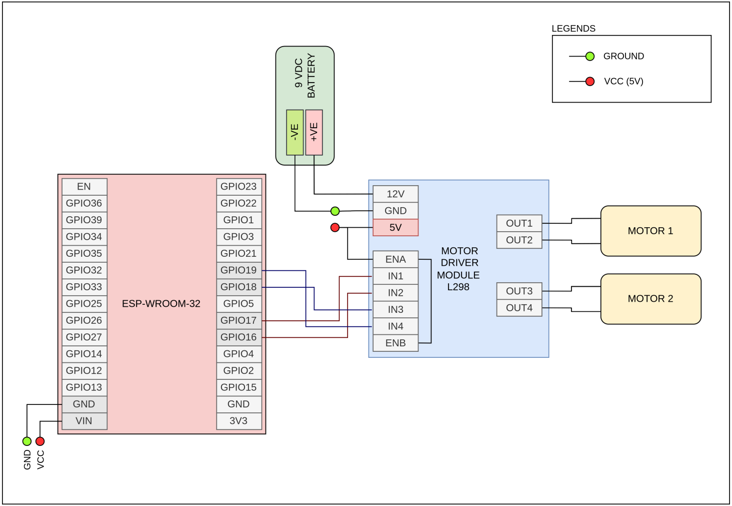


Figure 6: UGV Navigation using Android phone (ESP32)(Manual and Speech)

ESP32 Based Application-3

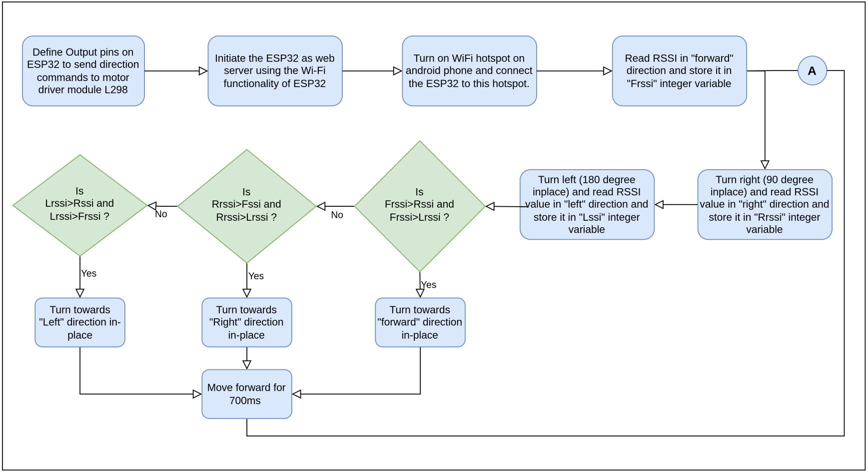


Figure 7: UGV beacon tracking

ESP32 Based Application-4

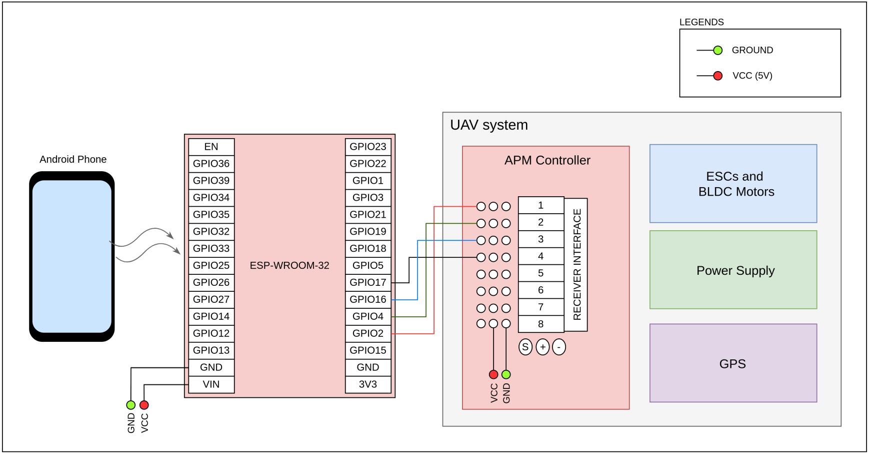


Figure 8: UAV Navigation using ESP32 and Android phone

Vaman Based Application-1

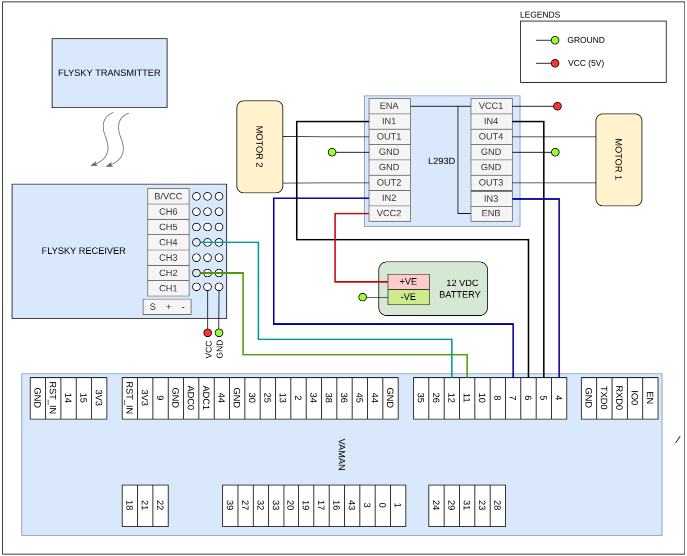


Figure 9: UGV Navigation using Fly-sky transmitter &

Vaman Based Application-2

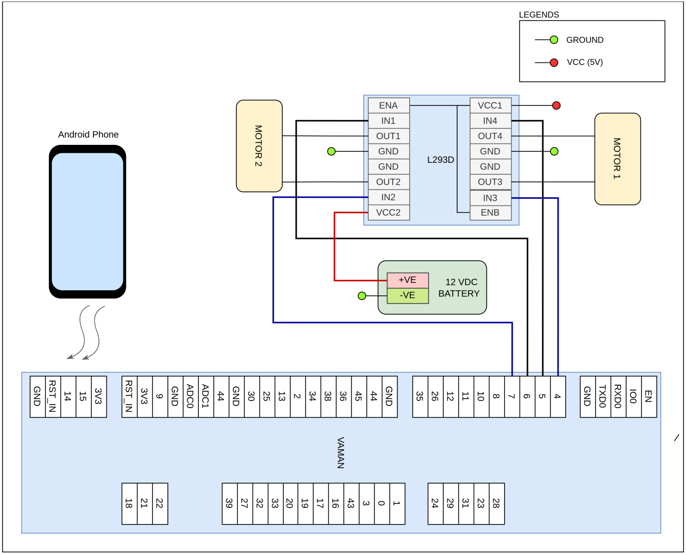


Figure 10: UGV Navigation using Android phone (Vaman)

SATCOM for UAV Communication

- Satellite Communication (SATCOM) can be used for UAV for remote access/control as well as transmit and receive data without requiring it to return to the operator.
- Convinced by the potential of 5G, Satellite industry has shown increased interest and participation in 3GPP to integrate satellite infrastructure with terrestrial network of 5G
- Figure 11 shows a typical 5G terrestrial network:

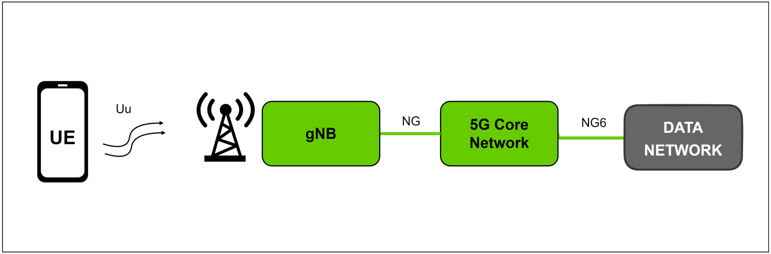


Figure 11: Conventional 5G-NR system

SATCOM for UAV (Continued)

- There are scenerios, where our user equipment (UAV in our case) may not be reachable from its nearest gNB (for example in remote areas like forests, mountainous terrain, etc).
- SATCOM provides a non-terrestrial network infrastructure, enabling communication with such remote devices.
- Latest advancements in the satellite communications have overcome previous constraints.
- New generation of Low Earth Orbit (LEO) constellations have lessened satellite communications latency
- Technological improvements in Geostationary (GEO) satellites have provided high throughput and increased the reliability of GEO satellites.

Transparent satellite based NG-RAN architecture

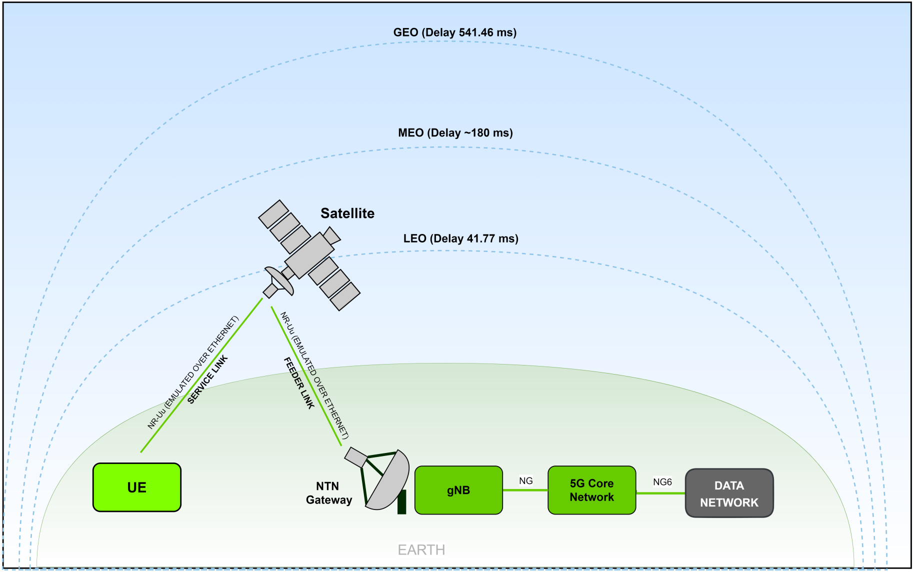


Figure 12: Transparent satellite based NG-RAN architecture

Set-up for Demonstration of SATCOM

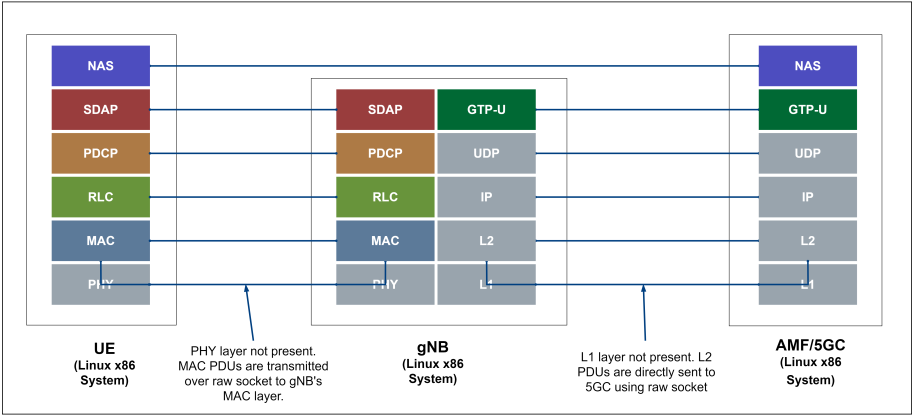
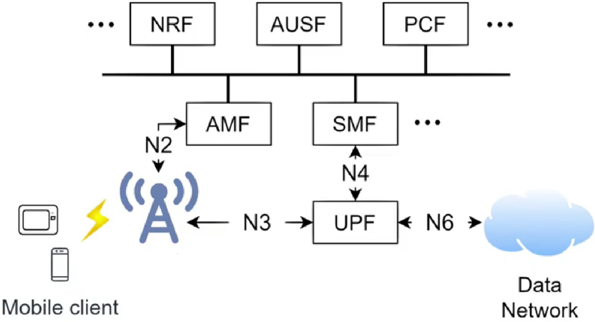


Figure 13: Set-up for Demonstration of SATCOM

Free-5GC Environment

- The Free5GC project is an open-source initiative for mobile core networks of the fifth generation (5G) aimed at construction of the 5G core network (5GC) as described in 3GPP Release 15 (R15) and futher.
- It our setup it acts as our 5G core network and runs on a linux-x86 system to provides services to the UE via the gNB as defined by the 3GPP specifications.



Compilation and Execution at gNB

- Exporting environment variables for DPDK (Data Plane Development Kit consists of libraries to accelerate packet processing workloads):

```
export RTE_SDK=<path to DPDK folder installed>
export RTE_TARGET=x86_64-native-linuxapp-gcc
```

- Loading Huge pages:

```
sudo su
echo 4096 > /sys/kernel/mm/hugepages/hugepages-2048kB/
    nr_hugepages
exit
```

- Compiling and running the gNB App:

```
cd ~/Documents/simran_wsp/bs_working/review-bs/5gnrps/src/
    gnbapp/test
make clean
make static -j10
sudo ./gnbapp enp1s0 -- -diersg
```

Compilation and Execution at UE

- Exporting environment variables for DPDK (Data Plane Development Kit consists of libraries to accelerate packet processing workloads)
- Loading Huge pages
- Compiling and running the gNB App:

```
cd /home/greyteal/Documents/simran_wsp/UE/5gnrps/src/ueapp/
    test
make clean
make CPUSOC=1 JSON=1 -j10
sudo ./ueapp enp1s0 0 -- -dierns
```

- Creating a tunnel interface for PDU session:

```
sudo ifconfig tun00 10.60.0.1 up
sudo ip route add 192.168.134.224 dev tun00
```

- Check the PDU session using ping (5 packets):

```
ping -I tun00 192.168.134.224 -c 5
```