## 3.2.5 Detection Subsystem

The main aim of the detection subsystem is to create a field vision for the teleoperator. Since this subsystem is the only source of information for the teleoperator, it must gather all the important data around the robot. This system must gather lots of visual data with minimum delay.

To design most suitable solution for detection unit, we decided to use a on board camera attached to the robot. Since the video data contains most of the valuable information the teleoperator needs, we decided to use an onboard camera for visual input. After obtaining the video input from the camera, the current frame of video is directly transferred to the telecontroller subsystem by communication subsystem without any processing. The transferred data is streamed on an LCD screen for telecontroller.

Due to the competitive nature of this project, the delay between teleoperator and visual of playfield must be as short as possible. To decrease delay time caused by detection subsystem, this system directly sends the raw video data without any processing. The block diagram and flowchart of detection subsystem can be observed in Figure 3.2.5.1. and Figure 3.2.5.2. respectively.

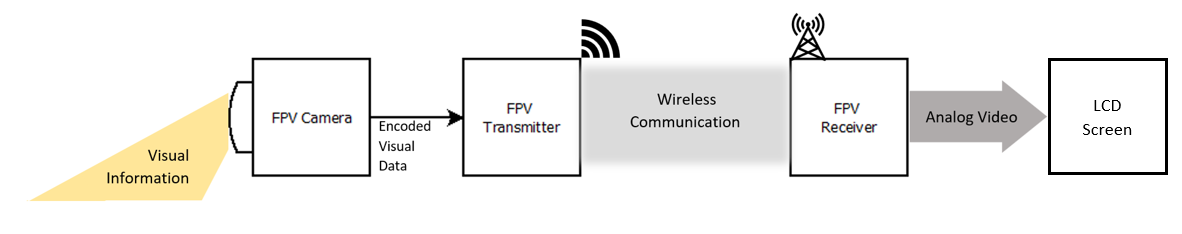


Figure 3.2.5.1: Block diagram of Detection Subsystem

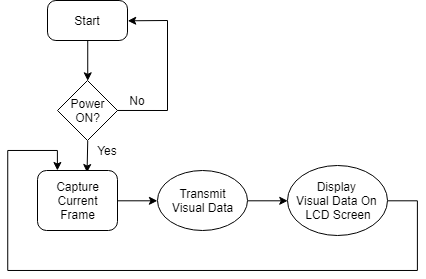


Figure 3.2.5.2: Flowchart of Detection Subsystem

As observed in figures provided, in the first step camera captures the real time visual information about the playfield. This visual information is encoded for the transmission and send directly to the receiver via RF channel. The transmitted information does not have any processing on it, due to the timing issues. The received real time video data is decoded and converted to the AV (Analog Video) format which is supported by our screen.

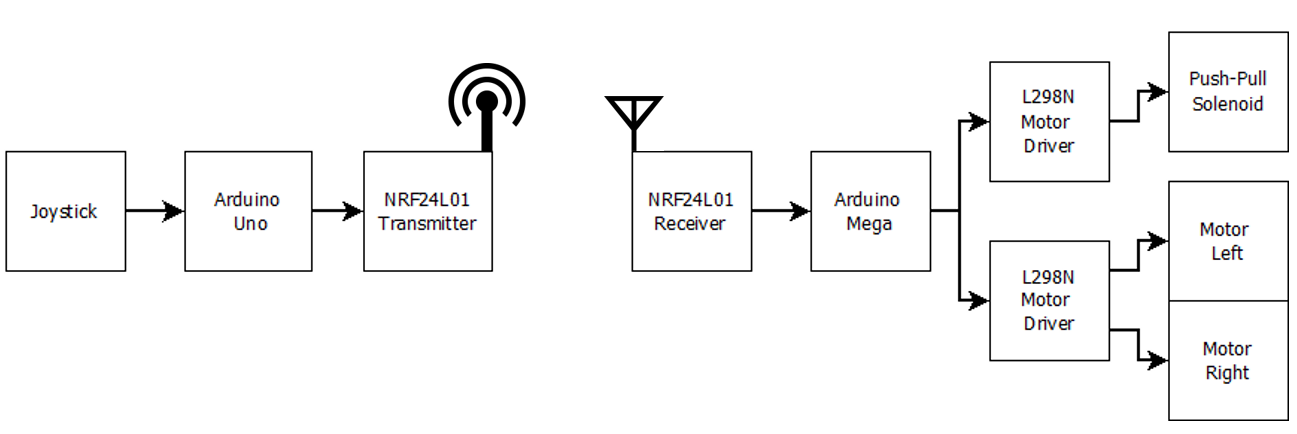
For the data transfer we use AKK KC03 800TVL NTSC Switchable Camera Module, shown in Figure 3.2.5.3, since it has a built-in transmitter, which we are going to use in communication subsystem. This camera module provides high quality video output for the telecontroller.

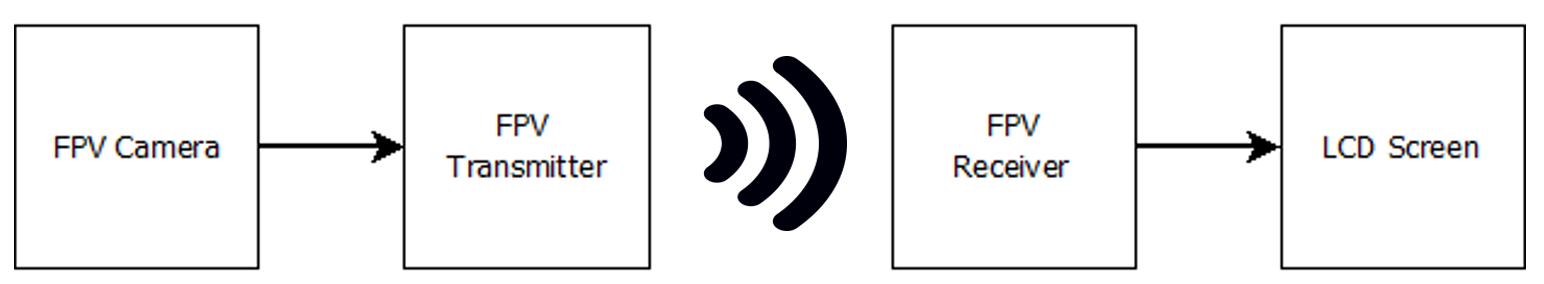


Figure 3.2.5.3: AKK KC03 800TVL NTSC Switchable Camera Module 600mW FPV Transmitter [11]

# 6 Compatibility of subsystems

In the success of a system, compatibility of subsystems has a crucial role. Any error in the between subsystem end up with serious performance issues or even nonoperating system. In our system, there are two routes for information flow. These routes can be named, command transmission and visuals transmission, which is shown in Figure 6.1 and Figure 6.2 respectively.

  
Figure 6.1: The path of information flow in command transmission

  
Figure 6.2: The path of information flow in visuals transmission

As shown in the figures provided, this system designed to have two non-interacting information paths This separate design makes us able to implement these paths without any compatibility issue. If we investigate the signals in the path of command transmission:

* The analog signal created by joystick is read by analog pins of Arduino Uno.
* The processed command information is sent to NRF24L01 transmitter module with a protocol written by the developers of NRF24L01 for Arduino. Since this protocol is specially written for module and Arduino communication, fully compatible for our design.
* In this step wireless signal is send from NRF24L01 transmitter to NRF24L01 receiver. As these modules are identical and designed to be used together they are totally compatible.
* The received information is decoded by the protocol, stated in transmit step, in the Arduino. Since this protocol is specially written for module and Arduino communication, fully compatible for our design.
* The final command needs to be sent to motors. However, a direct interaction between Arduino and motors is impossible due to current and voltage limitations of Arduino. To solve this compatibility issue, we use L298N motor driver to make Arduino able to control motor with PWM signals.

If we investigate the signals in the path of command transmission:

* FPV camera records real time video and encodes this information into data packages by internal protocol. This packs can be directly sent to FPV transmitter since both components have hardwire connections and use same communication protocols.
* This encoded packs are transmitted to the FPV receiver. The transmission has no compatibility problem, since both transmitter and receiver are from the same module and designed to work together.
* The received data packs are decoded by internal algorithm in FPV receiver. This decoded data is converted into analog video format.
* Our LCD screen has a build in analog video input, which can stream the incoming video data without any problem.

## 9.1. Cost Analysis and Breakdown

**Motors**

Since the dimensions of the field is not too large, there is not so much space for our robot to freely move. In order to create precise movement motors of the robot must have a high torque and low RPM. Brush DC motor with gearbox is most suitable option among other option on both performance and price level. Hence, we use DC motor with gearbox for movement. With this decision we spent 22% of our budget on motors.

**Development boards**

In the market, development board options are countless. For every application, there are dozens of options with very large performance and price range. In our case, the need for computational power very low. On the other hand, speed is very critical which eliminates high performance complex boards like Raspberry Pi or Intel series. For our operation Arduino family is more suitable since, the design is mature, IDE is well designed and have a great community and prices in our range. Moreover, the clone series especially decrease the prices, which create a great advantage for us. Hence, we decided to use less capable Clone Arduino Uno for telecontroller and high-performance Arduino Mega for robot. With the introduction of clone modules, development boards only hold 7% of our budget

**Communication**

The quality and reliability of communication between robot and telecontroller has a crucial importance in the success of this project. To ensure the robot performs under in desired circumstances without any error, we set our communication budget as high as possible.

Video transmission is especially challenging due to size of the data. Despite its relatively high price, the video quality and transmission range are far better than other options. Hence, we choose FPV drone module among other options.

Command transmission is comparably easy, since we deal with smaller size command data. Hence, we prefer a cost-effective module, NRF24L01. For these two modules we spent 22% of our budget

**Power**

The power network of this project is carefully designed to powerup all the components until the end of the operation. Also, the weight of the batteries is another issue. We decided to use Li-Po batteries (one for each side) with step-down converters to create a lightweight solution within our budget. This power system cost us 19% of our budget.

Table 9.1.1: Revised cost breakdown

|  |  |  |  |
| --- | --- | --- | --- |
| **Materials** | **Quantity** | **Price per each** | **Total Price** |
| **Motors** |  |  |  |
| DC Motors | 2 | $10 | $20 |
| Gearbox | 2 | $5 | $10 |
| Encoders | 2 | $5 | $10 |
| L298N Motor Driver | 2 | $1.5 | $3 |
| **Development Boards** |  |  |  |
| Clone Arduino Uno | 1 | $4 | $4 |
| Clone Arduino Mega | 1 | $10 | $10 |
| **Shooting** |  |  |  |
| Push-Pull Solenoid | 1 | $4 | $4 |
| **Communication** |  |  |  |
| FPV Drone Kit | 1 | $35 | $35 |
| NRF24L01+PA+LNA SMA | 2 | $4 | $8 |
| **Telecontroller** |  |  |  |
| Joystick | 1 | $0.50 | $0.50 |
| LCD screen | 1 | $30 | $30 |
| **Power** |  |  |  |
| 900 mAH Li-Po Battery | 1 | $14 | $14 |
| 2900 mAH Li-Po Battery | 1 | $20 | $20 |
| Buck Converter | 4 | $1 | $4 |
| **Structural** |  |  |  |
| Plexiglass Chassis | 1 | $6 | $6 |
| Standard Wheels | 2 | $2 | $4 |
| Ball Wheel | 3 | $1 | $3 |
| Cables & Connectors | nAn | $6 | $6 |
| 3D Printed Parts | nAn | $2 | $2 |
| Structural connections | nAn | $3 | $3 |
| **Demo Setup** |  |  |  |
| Dummy Robot | 1 | $2 | $2 |
| Balls | 2 | $0.25 | $0.5 |
| Play Field Walls | 3 | $1 | $3 |
|  |  |  |  |
| **Total:** |  |  | **$199** |

## 9.2 Power Distribution Analysis

In the development of battery powered projects, the power analysis has a crucial role. In our project we have a not one, but two battery powered parts to design and develop. Since the batteries are the only power source our robot and telecontroller has, to ensure proper operation for a predetermined time the power analysis must be done carefully.

In the first step of the power analysis, a distribution scheme for both parts are prepared. This connection schemes for robot and telecontroller can be observed in Figure 9.2.1 and Figure 9.2.2 respectively.

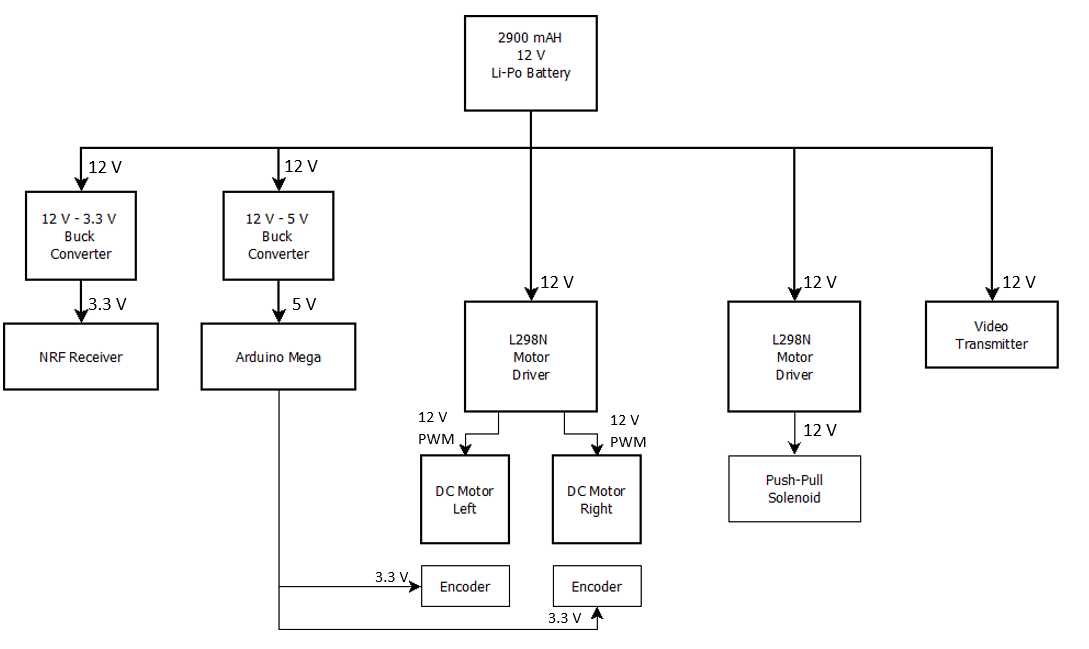


Figure 9.2.1: Power distribution scheme for Robot

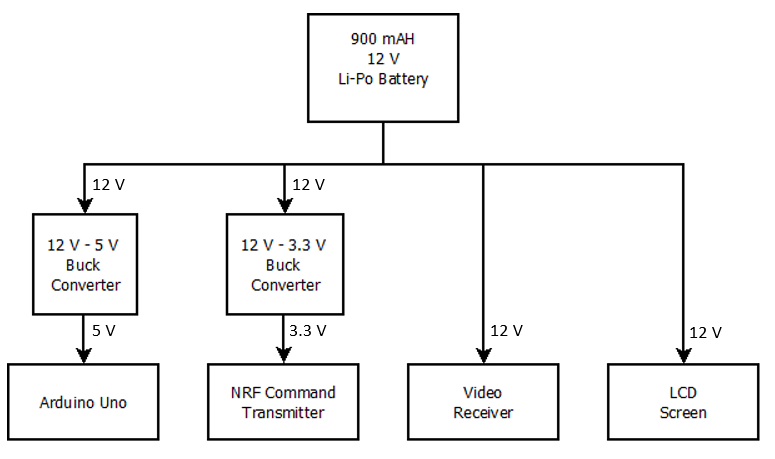


Figure 9.2.2: Power distribution scheme for telecontroller

In the total power consumption calculation of the system, we considered the worst-case scenario of each design element separately. Despite, this analysis method overestimates the power consumption, it creates a safety margin for us. In the case all the systems are active and at their full power, our system is going to be still fully functional. The power analysis of robot and telecontroller can be observed in Table 9.2.1. and Table 9.2.2. respectively.

Table 9.2.1: Power analysis of Robot

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Device | Quantity | Maximum Values per Device | | | Total Device Power |
| Current | Voltage | Power |
| DC Motor | 2 | 300 mA | 12 V | 3.6 W | 7.2 W |
| Drone Transmitter | 1 | 400 mA | 12 V | 4.8 W | 4.8 W |
| Push-Pull Solenoid | 1 | 250 mA | 12 V | 3 W | 3 W |
| NRF24L01 Receiver | 1 | 100 mA | 3.3 V | 330 mW | 0.33 W |
| Arduino Mega | 1 | 100 mA | 5 V | 500 mW | 0.5 W |
| **Total Power** |  | | | | **15.83 Watt** |

Table 9.2.1: Power analysis of Telecontroller

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Device | Quantity | Maximum Values per Device | | | Total Device Power |
| Current | Voltage | Power |
| Drone Receiver | 1 | 10 mA | 12 V | 120 mW | 0.12 W |
| Screen | 1 | 400 mA | 12 V | 4.8 W | 4.8 W |
| NRF24L01 Transmitter | 1 | 100mA | 3.3 V | 330 mW | 0.33 W |
| Arduino Uno | 1 | 100 mA | 5 V | 500 mW | 0.5 W |
| **Total Power** |  | | | | **5.75 Watt** |