South Asian Journal of Research in Engineering Science and Technology
Volume 4, Issue **1**, **January 2019**

www.sajrest.com
IXplore SAJREST ⓒ2019 ISSN 2455-9261
pp. **1--10**

EMBEDDED BASED PROGRAMABLE AUTONOMOUS MULTIROTORS SYSTEM

L.Karthi[1],P.Mathan Narayanan Samy[2] and S.Palanimurugan M.E.,(Ph.D)[3]

1,2 UG Scholar Department of Computer Science Engineering

3 Associate professor Department of Computer Science Engineering

EGS PILLAY ENGINEERING COLLEGE ,NAGAPATTINAM,TAMIL NADU

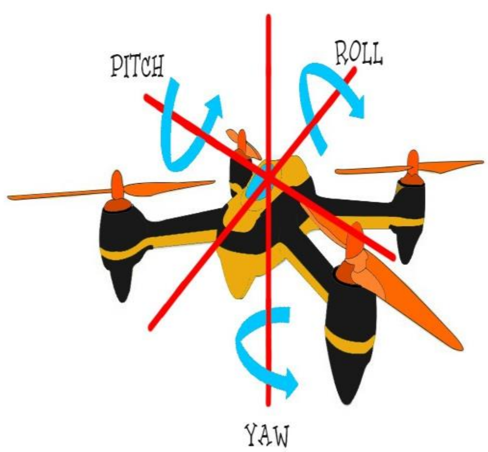
Received June 2008; revised December 2008

ABSTRACT. *Developing methods for autonomous landing of an unmanned aerial vehicle (UAV) on a mobile platform has been an active area of research over the past decade, as it offers an attractive solution for cases where rapid deployment and recovery of a fleet of UAVs, continuous flight tasks, extended operational ranges, and delivery the parcel are desired. We choose as four points like A,B,C,D automatically the flying on autonoumous using embedded sytemof programming lanugage. .In this work, we present a new autonomous landing method that can be implemented on micro UAVs that require high-bandwidth feedback control loops for safe landing under various uncertainties and wind disturbances*

Keywords: quadcopter; drone; Kalman filter; vision-based guidance system; autonomous vehicle; unmanned aerial vehicle; model predictive control; aerospace control

1. **Introduction.** A Quadcopter is a multicopter lifted and propelled by four rotors.In this project we aimed to build an Autonomous Quadcopter which can balance itself while flying.This quadcopter also consists of manual control system (transmitter-receiver) .For our project we decided to build the quadcopter frame in a simpler way having symmetrical four arms on which a motor with a propeller is mounted on every arm.In these quadcopter is flying on autonomous to check the condition which. Is our project like a just we point some object.it is automatically point and cross our object .automatically .it will landing off .In embedded based program show the vaious difference.

Each pair of opposite rotors is turning the same way. One pair is turning clockwise and the other is anticlockwise



Drones are obtained by changing pitch, roll and yaw angles of the AR.Drone

Assisted control of basic manoeuvres

Usually quadrotor remote controls feature levers and trims for controlling UAV pitch, roll, yaw and throttle. Basic manoeuvres include take-off, trimming, hovering with constant altitude, and landing. It generally takes hours to a beginner and many UAV crashes before executing safely these basic manoeuvres.

Thanks to the AR.Drone onboard sensors take-off, hovering, trimming and landing are now completely automatic and all manoeuvres are completely assisted.

Thanks to the AR.Drone onboard sensors take-off, hovering, trimming and landing are now completely automatic and all manoeuvres are completely assisted.

• When landed push take-off button to automatically start engines, take-off and hover at a pre-determined altitude

•When flying push landing button to automatically land and stop engines.

• Press turn left button to turn the AR Drone automatically to the left at a predetermined speed. Otherwise the AR Drone automatically keeps the same orientation.

• Press turn right button to turn the AR Drone automatically to the right. Otherwise the AR Drone automatically keeps the same orientation.

• Push up button to go upward automatically at a predetermined speed. Otherwise the AR Drone automatically stays at the same altitude.

• Push down to go downward automatically at a predetermined speed. Otherwise the AR Drone automatically stays at the same altitude.

A number of flight control parameters can be tuned:

- altitude limit
- yaw speed limit
- vertical speed limit
- AR.Drone tilt angle limit
- host tilt angle limit

Proposed Class Outline and Technical Requirements

The class can be conducted in a general laboratory environment available at the university, but the flight tests should be conducted either in a spacious enclosed facility or on the designated test stands, in order to avoid any possible damage done to the students by the rotating propellers. In addition, the flight in tight environments may result in a sufficient damage to the device done as a result of the contact between the obstacles and the poorly-calibrated or inaccurately controlled UAV.

technical requirements for the quadcopter they are supposed to create during the semester: desired vehicle payload, desired horizontal flight speed, and operational time (time that a UAV can spend in the air without recharging the batteries). Satisfying all of these three requirements is not a trivial task, since, for example, one needs to have larger motors/propellers in order to increase the payload, which can negatively affect the operational time. The requirements should be designed in such a way as to give the students a chance to improve on some or even all of the requirements by an efficient design, optimizing control system, or any other means

The students can be then divided into the teams of 3-5 people each, and the objective of every team will be to design and implement a quadcopter by the end of the semester. The duration of a weekly class can be between 2 or 3 hours, with additional homework assignments, if required.

The most important steps in the proposed class can be summarized as follows:

- Modeling, design, and assembly of hardware (frame, motors, electronics)
- Motor control system design and implementation
- using embedded system on programming is control by the drone on pc
- Remote control communication implementation
- Final experimentation and performance evaluation

MECHANICAL DESIGN

- Design the size of the future quadcopter based on the required specifications (maximum payload, desired flight time, etc.). The students may be given a choice between the existing propellers of various sizes and shapes – this may affect the size of the frame.

- Design the frame in order to incorporate all control electronics and motors, assuming the dimensions of all the parts and components to be known.

- Verify the frame design by installing the main electronics board, motors with drivers,software, sensors, and battery on it.

SENSOR INTEGRATION

- Gyros/accelerometers

- Tilt sensors

- magnetic sensor

established via electronic speed controllers (ESCs)without the use of motor velocity sensors. However, for the sake of gaining practical skills, the students may be asked to design a closed-loop control system with a velocity sensor feedback.

If needed for any sensor,we use it,the names is given below

- GPS (for path planning and following)

- Laser range-finders (for map building)

- Altitude sensors (to maintain constant altitude during the flight)

- Video systems, which may include stereo cameras (for 3-D object identific

CONTROL SYSTEM AND PROGRAMMING

The main objectives for the students develop the control system Drone is control by the programable embedded system When our drone is using some language for ex c,c++,python like this.in these coding feed by the embeded hardware.

The embedded is control the drone using to take of autonomously.Like it is delivered automatically

Autonomous delivery system

AERIAL VEHICLE

3.1. Propulsion system

The multirotor uses 15_5.5 inches propellers to provide sufficient lift. The propellers also work most efficiently at mid throttle ranges i.e. 50-55% throttle. The multirotor is designed such that at 50% throttle the weight of the multirotor is equal to its thrust.

These propellers are driven by Tigerc MN4010 [12]. They have comparatively low current ratings between 4.2A and 14.6A at 22.2V in 50% to 100% throttle range respectively. They operate at a maximum current rating efficiency of 84% between 3-8A at a temperature of 52_C, ensuring longer flight times.

T-motor 40A ESCs are used to drive the motors. These ESCs are recommended for the Tiger motors for 4S-6S configurations. The ESCs are flashed with Simon K firmware and calibrated through the Pixhawk to avoid de-sync during flight. The ESCs are connected to the motors using bullet connectors which can handle currents up to 50A.

3.2 State Estimation and Control

The state estimate of the multirotor is obtained by FCU's Inertial Measurement Unit's (IMU). The fusion is carried out by the ECL implementation of the EKF state estimation algorithm. The algorithm's parameters have been tuned on observations derived from manual flight and errors in measurement of sensor data inferred by analysis of flight logs. Further, the state estimation is converted to corresponding PID to control the multirotor. This is implemented in the internals of the PX4 v1.7.3 flight stack.

3.3Flight Termination System

Even the best engineering designs are bound to fail in certain hostile conditions in the eventof any system malfunction. In a multirotor there are several sub-systems and failure of any one of them could be precarious. To ensure safety of the user or individuals around this multirotor, we have introduced several safety measures into the system. If the autonomous or path planning system fails, the multirotor is programmed to return to its starting position and land. In the event of failure of positioning system, the aerial vehicle shifts to manual Radio-Controlled mode. For any other breakdown, we can disengage the power to the Electronic Speed Controllers (ESCs) and motor by using a radio-linked switch (kill switch) which works independent of all other communication links and computing system.

This will immediately terminate the flight.

3.4. Power Supply Unit

The system is powered by Lithium Polymer batteries rated at 4000mAH at 22.2V. Further, the individual subsystems receive their rated voltage from a custom-designed voltage divider circuit. Considering the current rating of the batteries and the estimated payload, the theoretical upper limit of the flight time with this battery and configuration of the system is about 20 minutes.

4. SYSTEM ARCHITECTURE

4.1. Overall System Architecture

The system consists of 2 cameras, a forward-facing stereo ZED camera and a downward-facing wide-angle camera. The system initially uses the ZED camera to scan the RoI to detect any bots that are headed in the wrong direction. The direction of the bots is then corrected. Further, to interact with the bots, the wide-angle camera is engaged in order to exercise fine control over the interaction. The interactions are then repeated until the bot reaches the green line and this process is repeated for all the bots. The bot detection sub-system interacts with the navigation architecture to command the multirotor robot to perform the interactions. The navigation module abstracts the vision sub-system from theintricacies of the low-level communication The interactions are accomplished by the navigation architecture, which in turn sends low-level commands via MAVLink v2 protocol to the FCU.

4.2. Navigation Architecture

The LIDAR sensor is used to detect the presence of obstacles in the environment. It can give a 2D snapshot of distances of various obstacles at a point in time.
The LIDAR sensor's data is consumed by the Lidar Node, which is a software module in the form of a ROS Node, which provides a wrapper that abstracts the low-level details of serial communication that is performed to obtain data from the LIDAR sensor, and in turn provides the readings of the LIDAR sensor in a programmer friendly manner over ROS strongly typed messaging queues.Page 8

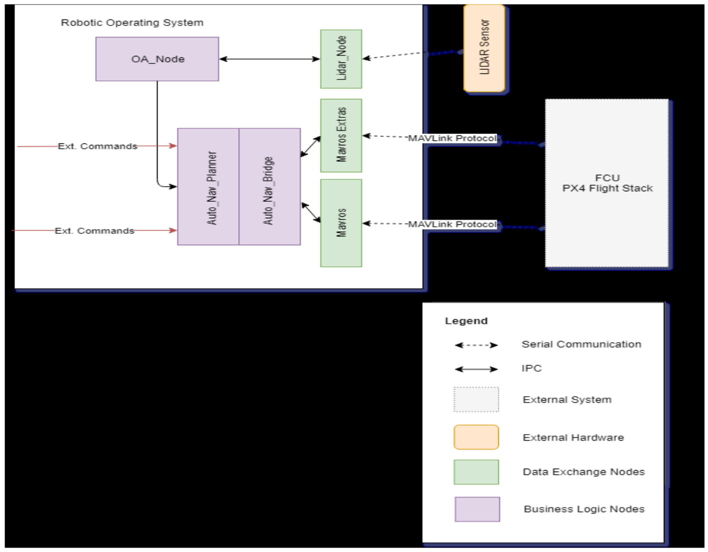


Fig. 3: Navigation System Architecture

The OA Node is the module responsible for obstacle detection. It uses the Lidar Node to obtain readings from the LIDAR sensor and performs analysis and computation over multiple instances or snapshots of LIDAR readings to detect obstacles and in turn provides details of obstacle presence. The node also performs data filtering. It restricts the distance to accurate range of the lidar, this parameter is configurable by the user.
The Mavros and Mavros Extras are ROS Nodes that abstract details of communication over serial or WiFi link with the MAVLink protocol for communication with the FCU.
The FCU receives messages over MAVLink that contain information and commands that it must follow. The FCU runs the PX4 firmware that runs on top of NuttX.
The Auto Nav Bridge node is the module responsible for low level communication with the FCU via MAVLink. It also takes care of publishing rates requirements from the MAVLink protocol and the PX4 Firmware. It also plays crucial role of continued publishing the current point on failure of planner node.
The Auto Nav Planner node is the module responsible for taking high level decisions to enable the aerial robot to navigate to the provided destination coordinates without colliding or touching the obstacles. It interacts with the OA and Bridge nodes to accomplish this. This module is the entry point for clients/users to use the system. It acts like a facade to the system, hiding its internal complexities and providing a simple interface for the client to use the system.

4.2.1. Enhancements over previous architecture:

This section will detail enhancements over the previous iterations of the system architecture.[13] The goal is to enhance existing features and develop new features for more stable and consistent navigation as well as object tracking. We have shifted our implementation for the navigation module from the optical flow(which gives inconsistent values) to using external odometry such as the ZED Camera which provides position coordinates using Visual Odometry[14]. Since external odometry is used in place of internal IMUs to track the multirotors position and since the external odometry is not directly connected to the Pixhawk, the Pixhawk is unaware of its position and hence navigation by publishing setpoints to the Pixhawk is no longer possible. The navigation code decides the next setpoint to go to, and by implementing PID in the navigation code, the required motion of the multirotor in the x,y and z axis to reach the next setpoint is determined.

These values are fed to the Pixhawk using Command Velocity Messages which then applies the appropriate thrust in the 3 axis to move the multirotor to its destination as decided by the navigation code. The new enhanced architecture uses the new ECL implemented EKF state estimator which fuses more data than LPE to estimate the robots altitude and is also compatible with the latest version of the MAVLink protocol (v2). The main enhancement made in Vision is the use of a bottom facing fisheye camera which is capable of detecting bots over a larger field of view compared to the camera used previously.

5. SYSTEM ROBUSTNESS

5.1. EMI/RCI

As discussed previously in Section 2.1.6, the aerial vehicle has a flight controller with embedded accelerometer, gyroscope and magnetometer. All these sensors are used alongside, in order to enhance the performance of the flight controller. These sensors are highly susceptible to external noise. The sampling frequency used by the flight controller unit is 1KHz, hence any other electromagnetic signal which has frequency less than 500Hz can corrupt the sensor values. The ESCs and motors draw high amperage of current at varying frequencies from about 1KHz to 300KHz and therefore it is extremely likely that the EM waves generated by them can corrupt the sensor data.

Additionally, the internal or external communication systems can introduce electromagnetic interference. The solution to this setback is electromagnetic shielding of the sensors and the wires carrying the signals or high amperage current to drive the motor. The EM shield is grounded with the body ground, which connects the universal ground of every circuit. By placing the IMU’s and ESCs strategically, the coupling can be reduced. Coaxial cables are preferentially used wherever possible. We have matched the impedance of every AC signal carrying conductor to reduce the reflection.

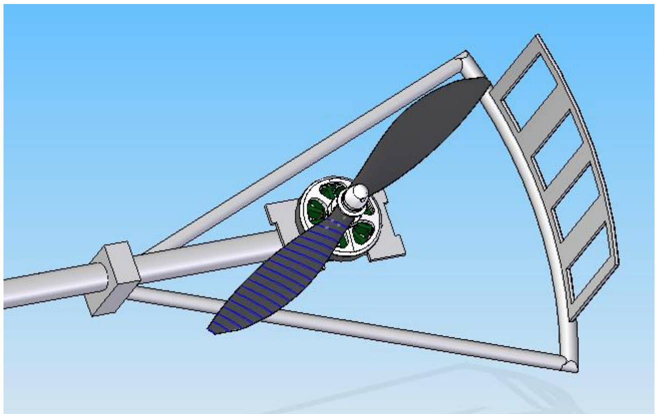


Fig. 4: Propeller guard

5.2. Vibration

Multirotor vibrations are one of the most critical problems faced during flight, take-off and landing. They need to be reduced in order to stabilize the multirotor. While taking off, landing and hovering at low altitude, ground effect occurs which causes considerable vibrations[15]. In order to avoid that, the take-off and landing time has been minimized and hovering will be done at a higher altitude, so as to have negligible ground effect. During landing, the vibrations have been overcome by introducing simple, cost effective bush dampers right below the landing gear. The four legs of the landing gear are all connected as one unit to reduce imbalance that causes vibrations as well. Rotating motors inevitably create vibrations that also adversely affect the sensors and visual display. We have used dampeners, currently foam, around motors and the other electronic parts to reduce these vibrations to a large extent. Along with the foam, there are rubber bushes under the flight controller unit to reduce the error in the sensor readings. Nut and screws turn loose due to vibrations and hence have been plated with thin sheets of foam to keep them intact. Our previous propeller guards were curved and closed structures. This caused the small amount of air flowing horizontally to hit them and induce vibrations. We have now introduced a Gated curved surface surrounding the propellers so as to allow maximum flow of air outwards.

5.3. Safety

While test flying the autonomous multirotor, all nearby team members stay 2 meters away from it at all times. In addition, the frame is always tethered with two ropes to prevent it from harming anyone or from crashing. To ensure the safety of the drone and the team members, we use prop guards attached to the arms of the multirotor. In cases of doubt the drone will stay grounded until the issue in question is found and resolved. Our prop guards took multiple iterations, these have been 3D printed using ABS lament given its strength, flexibility, machinability, and high temperature resistance. We wanted a design that would not interfere with the prop wash, but still be able to avoid the multirotor from destroying itself in the unfortunate event if collides with objects around it.

Conclusions. In this paper, an outline of a quadcopter design and implementation class is presented. After completion of the class, the students will gain very important skills of design and further practical implementation which can be extremely beneficial for the students as the future engineers and researchers in the fields of robotics, mechatronics, automation and control. the influence of various types of controllers on the overall stability of the inherently-unstable aerial vehicle. The proposed class is very flexible.

REFERENCES

https://www.researchgate.net/.../261061267_Quadcopter_design_and_implementation_a
https://www.dronezon.com/learn-about-drones-quadcopters/how-a-quadcopter-works-with-propellers-and-motors-direction-design-explained/

https://courses.ece.cornell.edu/ece5990/ECE5725_Fall2017_projects/Autonomous_Quadcopter_PublishOutput/Main.html

[1] S. Bouabdallah, “Design and control of quadrotors with application to autonomous flying” (Ph.D. thesis), École Polytechnique Fédérale de Lausanne, 2007.
[2] M.Dupuis, J. Gibbons, M. Hobson-Dupont, A. Knight, A. Lepilov M. Monfreda, G. Mungai, “Design Optimization of a Quad-Rotor Capable of Autonomous Flight” (Major Qualifying Project Report), Worcester Polytechnic Institute, 2008.
[3] M. Silberman and C. Auerbach, “Active Training: A Handbook of Techniques, Designs, Case Examples, and Tips (3rd Ed.)”, John Wiley & Sons, Inc, 2006.
[4] P. Pounds, R. Mahony, and P. Corke, “Modelling and Control of a Quad-Rotor Robot,” in Proc. of the Australasian Conf. Robotics and Automation, 2006.
[5] A. Bachrach, R. He, S. Prentice, and N. Roy, “RANGE-robust autonomous navigation in gps-denied environments,” in Proc. IEEE Int. Conf. Robotics and Automation, Kobe, Japan, May 2009.
[6] UAVforge: DARPA and SSC Atlantic collaborative initiative.url:http://www.uavforge.net