THE DEVELOPMENT OF A LOW COST AUTONOMOUS UAV SYSTEM

Meng-Tse Lee*, Wen-Ying Chang*, Cheng-Chen Yang*,

Kuo-Wei Lin*, Yi-Feng Tsai*, Chun-Rong Wu*, Fei-Bin Hsiao[#]

Institute of Aeronautics and Astronautics
National Cheng Kung University
Tainan 70101, TAIWAN, ROC

Keywords: *Unmanned Aerial Vehicle, System Integration, Autonomous System*

Abstract

Unmanned Aerial Vehicle (UAV) has proven their values and practicability in a variety of civil, military and scientific applications for many decades. With the aims of extensive usage of such UAV systems, one of major enabling-technologies is the automation of the system. This paper describes our attempt to create a miniature autonomous UAV in National Cheng Kung University (NCKU), including vehicle itself, optical payloads and a ground monitoring station in a low cost. Integrate essential hardware such as PC-104 single board computer, peripheral sensors, GPS and inertial measuring unit into a commercial R/C airplane. For the onboard software, the sensed-data and a proportional flight control (P-Control) program are processed on a real-time and multi-tasking operation system. This distributed software structure concept is highly modular to make the entire onboard system more flexible that can easily be reconfigured when necessary. In addition, a PC-base ground station is developed to display the real time in-flight information for the operators.

1 General Introduction

Unmanned vehicle system has shown its importance both in military and civil all over the world. The United States has became the first country that applies UAV into battlefields from

* Graduate Students
# Professor

reconnaissance Remotely Piloted Vehicle (RPV) in Vietnam War to lethal Unmanned Combat Aerial Vehicle (UCAV) in the war against terrorism. The use of UAV marks a corner stone in military history that the Air Force is now able to survey and even attack the ground targets from lower altitude without employing pilots in risks. For civil applications, the UAV also displays its potential for weather observation, communication relay, and optical remote sensing in routine patrol. The blooming applications in UAV urge more and more universities to start the researches in this field.

In year 2000, the School of Engineering Science in Simon Fraser University had developed an autonomous UAV called FireMite [1]. And the GTAR (Georgia Tech Aerial Robotics) team completed Level I goal at the 2001 International Aerial Robotics Competition, organized by the Association for Unmanned Vehicle Systems (AUVS). A MicroPilot MP2000 [2] is utilized for the requirement of autonomous flight [3].

The research activity of autonomous UAV systems has also arisen for recent years in the universities in Taiwan. RMRL (Remotely piloted vehicle and Microsatellite Research Laboratory) in Institute of Aeronautics and Astronautics of National Cheng Kung University is one of the major research laboratories in Taiwan for UAV development. The entire course of UAV system development in RMRL can be referred to the review paper [4].

The goal of this project is to develop an entirely autonomous aerial operation over 30

km away. To achieve this goal, an autonomous UAV and a base monitoring station are required. One of the mission requirements is defined as autonomously taking the aerial photography by an automatic camera on a pre-assigned spot beyond the visual range site.

In recent practical UAV applications in RMRL, the importance of beyond-visual-range (BVR) flight capability is revealed gradually. The BVR flight capability will make the aerial vehicle more useful for most of the missions in practice. There are two ways to achieve the BVR flight: remote controlling the UAV via instrument data/real-time video and autonomous flight by UAV itself. The latter one is adopted in the current study. To accomplish the goal for an autonomous flight is complex, which will combine both hardware and software in the system. The hardware aboard the aircraft usually consists of a computer, which serves as a heart of the autonomous flight, sensors, and some electronic devices. While on the ground segment, there are a laptop personal computer that is basically portable and wireless communicating devices. The integration of the hardware on UAV and the communications between airborne and ground system is important and a tough task.

2 System Overview

The high-level system architecture of a complete airborne/ground system is shown in Figure 1. The airborne system comprises the aerial vehicle and an onboard system. The ground system comprises a laptop PC and a radio modem to communicate with the airborne system.

To pack all the onboard hardware, a suitable vehicle with large interior capacity is necessary. A 1/5-scale miniature Cessna 182 is chosen for meeting the project requirement. The model is powered by a 2.5 hp gasoline engine, with a wingspan of 2 m and a fuselage diameter of about 0.19 m. This large interior capacity has the advantage of being able to facilitate all of the electronic components needed and the optical payload up to 5 kg totally.

Developing the onboard computing system on a PC base single-board-computer would be easy to establish and extend. The onboard computer (OBC) is developed under the QNX software, which is a real time operating system with multi-tasking and microkernel. Based on the fundamentals, the OBC can acquire data (GPS, INS, sensors, etc…), computing the flight control and navigation algorithms, command the optical payload, store and downlink data to the ground station in real time operation, etc.

A low cost and easy to move is our expectancy to the ground station. The ground station can real time display the status of the flying UAV including position, attitude and system health in visual instruments. It hence provides us more sense to realize the flight conditions of the UAV just like a pilot seating in the cockpit. The actual position can also be illustrated on an electronic map in the ground station. For the portable requirement, a laptop PC is the first candidate to be considered. The amiable, easy to access the price also reduces the budget of developing the ground system.
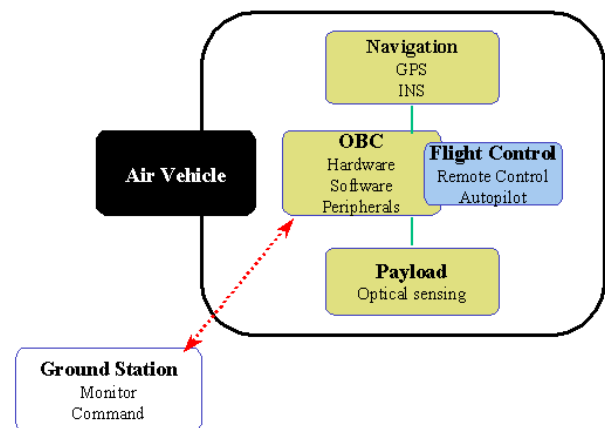

Figure 1: High level system architecture

3 Onboard Systems

3.1 Onboard Hardware

The major hardware architecture excluding the vehicle is shown in the left part of Figure 2. It is composed of an onboard computer, sensors, a servo controller, an RC switch, a wireless modem, and a power regulator.
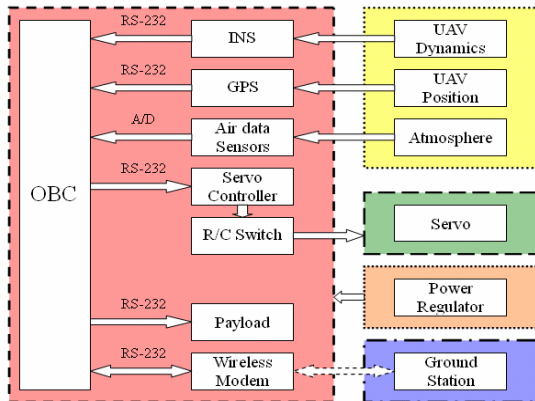

Figure 2: Onboard hardware architecture

The onboard computer illustrated in Figure 3 is operated based on a Single Board Computer (SBC) module, which has originally designed for industrial application and has the same dimensions as that of a standard PC-104 module (90 x 96 mm). It contains almost everything that a full-fledged computer requires in spite of its small size. An RS-232 module is added to increase the series ports in order to have more connections with other devices, and a DAS module is attached to serve as an A/D converter to acquire the data into the computer. The three modules (that is, computer, RS-232 and DAS) can be stacked together to provide an open and flexible platform for wide and advanced applications of UAV developments.
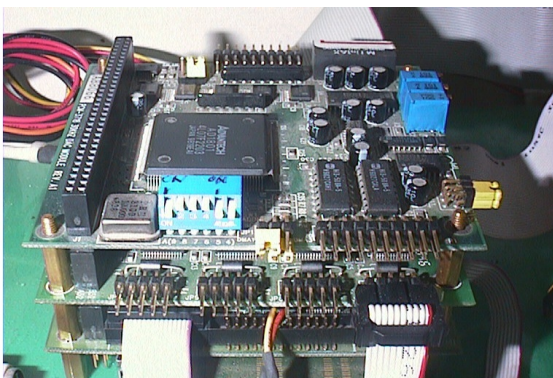

Figure 3: Onboard computer in a stack

The UAV employs two kinds of sensor: one is for air data sensing, and the other is for position determination. A GARMIN GPS 25LP is used to determine the position of the UAV and to provide the information for navigation as well [5]. The position accuracy in non-differential mode is 15 meters RMS in 1 second update rate. The air data sensors acquire the pressure altitude and airspeed information and output the signal into the computer through the A/D converter. The pressure sensor provides more stable altitude information than the GPS receiver does. It's very important for the needs of the autopilot control law and navigation algorithm.

To maintain the data link with the ground station, a Freewave wireless modem is utilized [6]. It can communicate with another modem in line of sight range up to 19 miles without any amplifier. To control the servos from the onboard computer, an interface that can convert the computer command to the PWM signal is necessary. A servo controller named SSC II provides the solution, which is an electronic module that controls eight pulse-proportional servos according to the instructions received serially at 2400 or 9600 baud rate.

Once the UAV take off and land manually, in that case, there must be a device to switch the function between the computer and the manned control. The self-made switchboard can switch four channels between the servo controllers and the RC receiver. The switch signal comes from one of the RC channels. Figure 4 shows the RC switch and the SSC II board in the left hand side and right hand side, respectively.
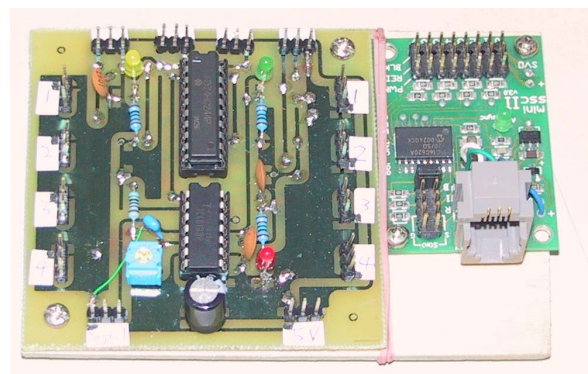

Figure 4: RC switch and SSC II

The various components form many kinds

of electric wires onboard. Each component has its own signal wire and power wire. It makes the preparation and installation complex before and after the flight. The idea to unify the wires is to make them into one. Here a 6-core phone wire bundle is used to serve this purpose. It combines the signal and power wire into one. An interface card is made as a bridge between the computer and its peripherals. There are 6 RS-232, 8 A/D, and 3 D/IO ports on the panel. All the ports are transferred to the phone sockets. Figure 5 shows the interface panel on the metal onboard computer case.
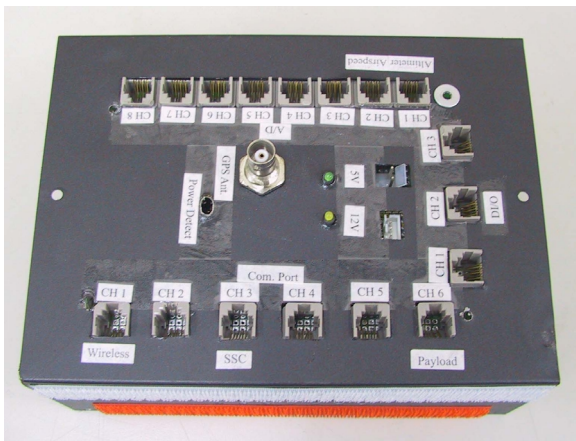

Figure 5: The OBC case with interface panel

3.2 Onboard Software

The overall software system is constructed based on the QNX operating system and writing in Watcom C language. It contains several individual processes, which "communicate" with each other via the unique IPC (Inter-process communication) function in QNX. Lin proposed the initial concept for the aerial data-acquisition system [7]. The software structure can be divided into four blocks and each block contains several processes, which execute similar functions.

Referring to Figure 6, the input processes are responsible for the data acquisition, such as position, attitude, airspeed and altitude of the UAV, and even for uplink data from the ground station. They collect useful data from sensors via the interfaces on the computer directly. The data from the sensors are not necessarily suitable for the needs of UAV.
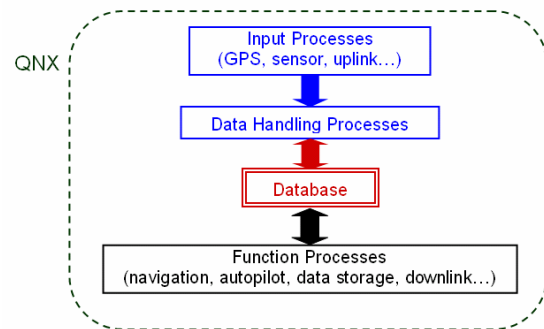

Figure 6: Process flow of software structure

Thus, the data would be sent to the handling processes for computing some transformations, so that other processes can recognize them. All the useful data and information are temporarily stored on the database process. It is like the role of random-access memory (RAM) in computer. Finally, all the mission execution is done in the function processes including navigation and autopilot algorithm, data storage, downlink function, and payload function. Of course, there are some processes sending final commands to control the hardware. The complete software architecture is illustrated in Figure 7.
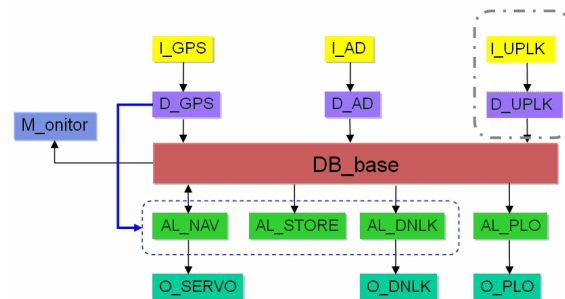

Figure 7: Software architecture

Each block in the Figure 7 stands for a single process and is a complete program. The I_XXX blocks represent input processes, while D_XXX blocks denote data handling processes and AL_XXX blocks the function processes. The M_onitor block is a monitor process that starts up all processes in a definite sequence at the beginning and monitors the system in the entire journey. Most functional processes except the payload process need to be triggered by the GPS time. This ensures all the processes can request the information from the database and execute the functions at the same time. The

conception of time unification is very important for the system, which is the only way that every process can get the same data at the same moment. Therefore, the data can easily be analyzed after the flight test.

The process of AL_NAV is the core of the autopilot. It contains the control law of autopilot and the navigation algorithm. There are three items that the process should be considered: what to climb/descend, where to go, and when to reach. The basic concept is that the database provides the information required for the autopilot, and then the program computes the parameters based on the control law to determine the appropriate angles that the control surfaces should act accordingly. Here, for simplicity the altitude holding only driven by the elevators and the direction holding by the rudder. The movement of the ailerons and the engine throttle are not included yet. The information of the deflections of the control surfaces is sent to O_SERVO process. The O_SERVO process transforms the angle information with the format that the servo controller can recognize and then sends to the servo controller through the serial port.

3.3 Optical Payload System

The mission of the payload system can be divided into two stages. First one is to catch the target image whose coordinates will be pre-assigned then triggered automatically by OBC once reaching the assigned target area. Second, the transceiver of the system will transmit the target image to the ground station with NTSC analog signal format so that the images can be displayed on a monitor in real time.

The CCD camera is set up on a servo platform with two degrees of freedom as illustrated in Figure 8. The conformation of the servo platform is similar to the gimbal type. It is designed to adjust the view of the camera on the ground target as one programmed in advance.

Through GPS and 3-axis gyroscope onboard, the UAV could know exactly where it is and its attitude. That information of the vehicle could be known and transferred into the

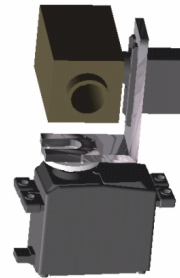OBC memory with the assigned format for coding and decoding.

Figure 8: The CCD camera with a servo platform

The driver of the servo platform will download the text file and determine the relation between the body frame coordinates of UAV and the line of sight of the CCD camera as shown in Figure 9. After that, the servo should move and rotate the camera to pin-point to the target through the two degrees of freedom movement. The servo platform can be designed not only for swinging the compensation but also tracking the target fixed on the ground. The software architecture of the driver is based on the coordinate transformation.
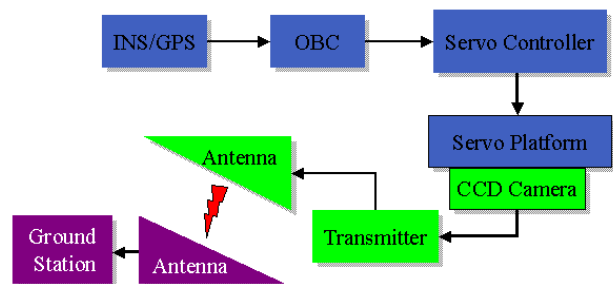

Figure 9 Flow diagram of optical payload system

4 UAV Flight Stability & Control

The flight parameters of the model Cessna 182 are listed in Table 1. Some of the dimensionless aerodynamic coefficients are directly derived from the airplane design handbook [8], the rest are calculated through DATCOM software. To simply the problem in this paper, one directly used the linear state equations for analyzing the automatic flight control system of UAV [9]. Because of the longitudinal- and lateral-directional linear state equations are decoupled, the analysis of control

laws in the longitudinal and lateral direction will be performed separately.

Table 1: The flight parameter of the model Cessna182

Flight Vehicle	1/5 scale Cessna 182
Speed V_1	72.9076(fps, 80km/hr)
Altitude h	1000 (ft)
Center of Gravity $(\bar{x}_{cg})$	0.25 $\bar{c}$
Wing Area S_w	6.3249 (ft^2)
Wing Span b	6.7388 (ft)
Wing Mean Chord $\bar{c}$	0.9386 (ft)
Weight W	22.0264 (lb , 10 kg)
I_{xx}	0.3642 (slug ft^2)
I_{yy}	0.6837 (slug ft^2)
I_{zz}	0.9465 (slug ft^2)
I_{xz}	-0.0442 (slug ft^2)

In lateral direction, the purpose of control is to track a reference command of azimuth (heading angle). By using the Matlab simulation program, the lateral simulation block diagram is shown in Figure 10. Two proportional gains are designed to achieve the aim of azimuth tracking. The state space equations of the lateral motion of the UAV shown in Figure 10 are given by Equation (1) and (2).

$$\dot{x} = Ax + Bu \quad \text{...........................} \quad (1)$$
$$y = Cx + Du \quad \text{...........................} \quad (2)$$

where

$$x = x_{lat} = [\Delta\beta \quad \Delta p \quad \Delta r \quad \Delta\phi \quad \Delta\psi]^T$$

$$u = u_{lat} = \begin{Bmatrix} \Delta\delta_a \\ \Delta\delta_r \end{Bmatrix}$$

$$A = \begin{bmatrix} -0.2374 & -0.0013 & -0.9926 & 0.4417 & 0 \\ -61.0914 & -15.4820 & 3.0011 & 0 & 0 \\ 14.8191 & -1.0996 & -1.1037 & 0 & 0 \\ 0 & 1.0000 & 0 & 0 & 0 \\ 0 & 0 & 1.0000 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0.1432 \\ 124.7371 & 8.2670 \\ -8.5858 & -17.4752 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$
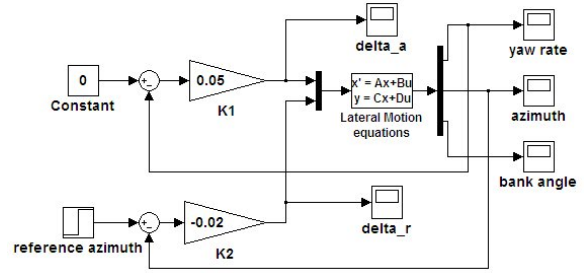

Figure 10 Simulation block diagram (lateral motion)

Lateral direction control surface

A positive aileron deflection means that the right hand side (back view) flap of the wing deflects upward and the left side flap deflects downward and will produce a positive roll moment. A positive rudder deflection will produce a negative yawing moment. For the control system design, the control surface can be shown as following:

$$\delta_a = -0.05 \times r \,(\text{degree})$$
$$\delta_r = -0.015 \times (\psi_c - \psi) \,(\text{degree})$$

where r denotes yaw rate (deg/sec), ψ_c reference azimuth command (deg), and ψ azimuth obtained and calculated by the GPS data.

Longitudinal feedback control system

For the mission that the UAV will be maintained in a constant altitude of 1500 ft, the longitudinal feedback control system for flight stability is designed according to Figure 11. There are two feedback loops in the control system, the inner loop feedback flight-path angle (γ) and the outer loop feedback altitude (h), so that there are two proportional gains to be designed to perform the altitude holding.
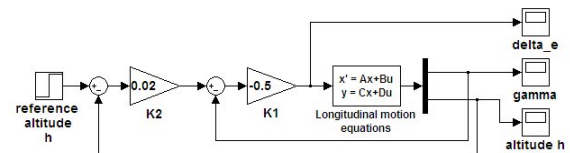

Figure 11 Simulation block diagram (longitudinal)

Longitudinal Equations of Motion:

$$\dot{x} = Ax + Bu \quad \text{................} \quad (3)$$
$$y = Cx + Du \quad \text{................} \quad (4)$$

where

$$x = x_{long} = [\Delta V \quad \Delta\alpha \quad \Delta q \quad \Delta\theta \quad \Delta h]^T \quad u = u_{long} = \{\Delta\delta_e\}$$

$$A = \begin{bmatrix} -0.0475 & 10.0480 & 0 & -32.2000 & 0 \\ -0.0065 & -3.5163 & 0.9726 & 0 & 0 \\ 0.0113 & -40.4848 & -5.8905 & 0 & 0 \\ 0 & 0 & 1.0000 & 0 & 0 \\ 0 & -72.9076 & 0 & 72.9076 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} -3.3493 \\ -0.3265 \\ -66.5256 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

After calculation, the gains K_1 = -0.5 and K_2 = 0.02 are chosen, then the responses with the reference command *r=unit step* are shown in Figure 12 and Figure 13.
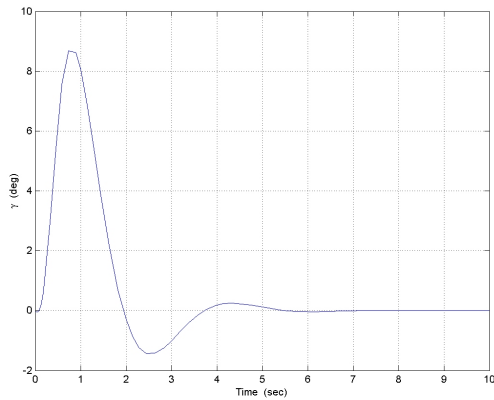
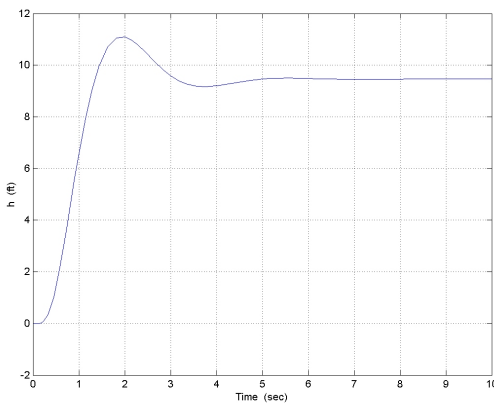

Figure 12 Flight path angle $\Delta\gamma$ versus time

Figure 13 Altitude *h* versus time

Longitudinal direction control surface

The deflection angle of elevator δ_e is positive when the elevator deflects downward with the aircraft pitch down, and

$$\delta_e = K_1 \times (K_2 \times (h_c - h) - \gamma) \times \frac{180°}{\pi} \quad (\text{degree})$$

where:
K_1 = -0.5
K_2 = 0.02
h_c = reference altitude command (ft)
h = altitude of UAV feedback by altitude meter (ft)

$$\gamma = \tan^{-1}\left(\frac{\dot{h}}{\sqrt{\dot{P}_N^2 + \dot{P}_E^2}} \right) \quad (\text{rad})$$

where $\dot{P}_N$, $\dot{P}_E$ and $\dot{h}$ are the velocities in northern, eastern and upward direction on NED frame system obtained from the GPS receiver.

5 Ground Station

The development of the ground station is mainly based on the LabVIEW software [10]. In this research, the LabVIEW software is chosen as the platform owing to its object-oriented and graphical programming environment. Comparing with the traditional line-by-line scripts interface, to construct a data flow chart in LabVIEW instead of scripts coding program is much easier. In addition, with the advantages of graphical and object-oriented functions, the ground station can be modified in a short time hence function module can be added or removed easily without changing the system architecture.

Table 2 shows the communication protocol between the airborne vehicle and the ground station. This protocol needs to be identically defined before the data transmission by request. The telemetry frame is led by the string of "$DL" and ended with an EOF (End-Of-File) character in the tail. For the recent experimental application, the total data length has 83 bytes, which include 3 bytes of head string, 73 bytes of telemetry data, 1 byte of EOF, 4 bytes of blank, 1 byte of carriage return, and 1 byte of change line. Once receiving the telemetry data, the leading character "$DL" will be striped away from the entire frame and leaves the remainder for further processing. The remaining data will then be converted into different sets of flight information and then displayed on the virtual instrument panel in both graphical and numerical forms on the screen of the laptop. Meanwhile, if the checksum error was detected, the system will immediately give a warning

with the flashing light and beeping sound to notify the operator. The decoding process of the telemetry data is illustrated in Figure 14.

Table 2 Format of the downlink frame

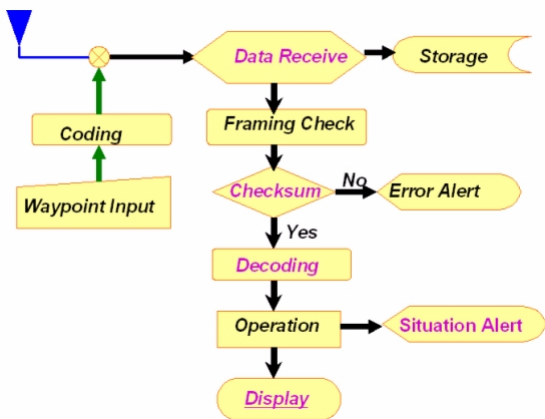| Item	UTC	Numbers of GPS	Latitude	N or S	Longitude	E or W	GPS altitude
Length (byte)	4	1	4	1	4	1	4
Item	GPS speed	Heading	Payload1	Payload2	Payload3	Payload4	Altitude
Length (byte)	2	2	1	1	1	1	4
Item	Air speed	Pitch	Roll	Yaw	Power supply	Regulated 12V	Regulated 5V
Length (byte)	4	4	4	4	4	4	4
Item	Aux.1	Aux.2	Aux.3	Checksum	Total		
Length (byte)	4	4	4	1	72		

Figure 14 Decoding processes of telemetry data

The main function of the ground station is to monitor the airspeed, attitude, altitude, vertical speed, heading, position, angular rates, and payload situation. Moreover, the position of the UAV will be mapped into an electric map; the caution light will be kept on to warn the operator when each flight speed, altitude and power supply is abnormal during its cruise phase. The waypoint uplink is the next significant function to the UAV, and therefore user can change the waypoints in-flight simply by key-in the new waypoint coordinates from laptop's keyboard. The monitor panel showing the visual instruments and numerical data is demonstrated in Figure 15.

Figure 15 Monitor panel of ground station

6 Tests and Results

The overall system tests can be divided into two major parts: ground phase and flight phase. The basic procedures of the system test can be referred to Figure 16.
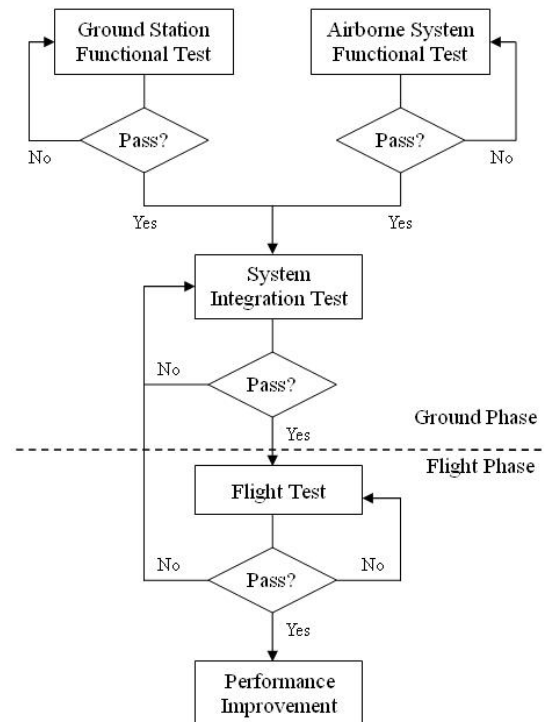

Figure 16 System test procedures

6.1 Ground Phase

The UAV has to be well tested on the ground before the flight test proceeds in the sky. These tests include individual sub-system level functional test, endurance test, and integration test of both airborne and ground system. The functional and endurance test ensures that the airborne and ground system can execute every function according to the requirements, which is followed by the integration test.

The integration test makes sure the entire system in regular status. The wireless data transmission test has revealed that the data delay in the ground station propagates with time increasing at the very beginning. The problem was solved soon, and the data can be synchronized both in airborne and ground system. Finally, the dynamic ground test on a car was implemented. The result shows that the ground station can well monitor and display the receiving data in real time.

6.2 Flight Phase

After the ground test, the first flight test was put into practice. The major purpose of this very first flight test is to ensure if the onboard computing system and the real time data link is accomplished under the flight condition. In this flight, the UAV crashed into the ground within 30 seconds because of the electromagnetic interference (EMI) problem. Fortunately, no obvious damage was found in the peripherals onboard but the airplane destroyed. The data recorded in the ground station reveal the altitude, airspeed, and other status of the UAV before crash as shown in Figure 17.
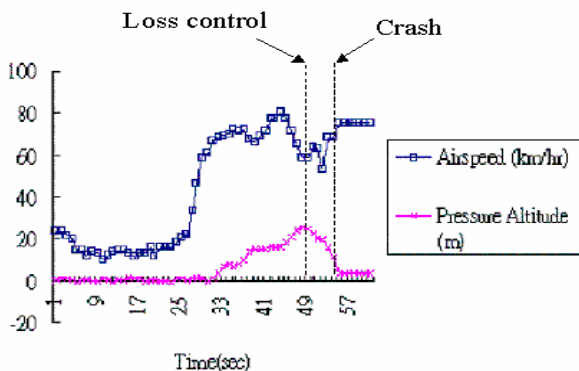

Figure 17 Altitude and airspeed versus time

From the lesson learned in first flight test, during the reconstruction of the aircraft and re-evaluation of numerous ground tests, the EMI test is carefully conducted to ensure that the electrical devices and RC receiver do not influence with each other, and the communication between the onboard computer and the ground station will not be influenced by other factors.

On May 13, 2002, the second flight test, the UAV proceeded again flying into the sky successfully and transmitted data to the ground station with many useful data, such as GPS, altimeter, etc, and the optical sensing payload (OBC controlled camera) perfectly work as well.

Figure 18 depicts the ground station screen snapshot of real-time 3D flight trajectory in the second flight.
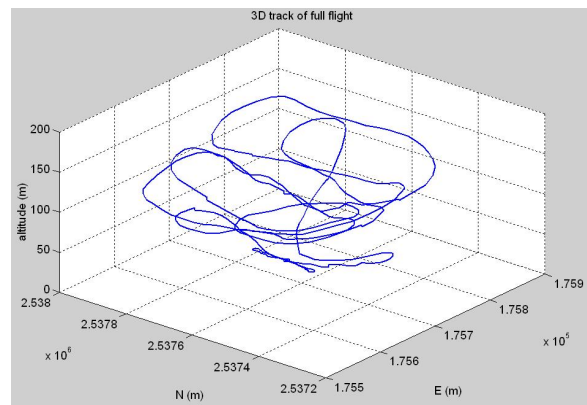

Figure 18 The full 3D track of the 2nd flight

Figure 19 illustrates the GPS altitude of the flight test. And the interrelated velocity variation is depicted in Figure 20
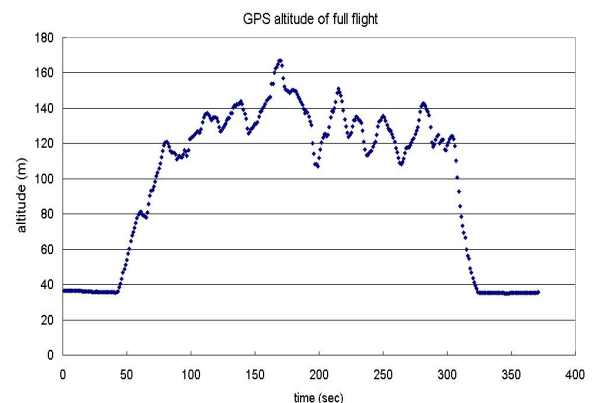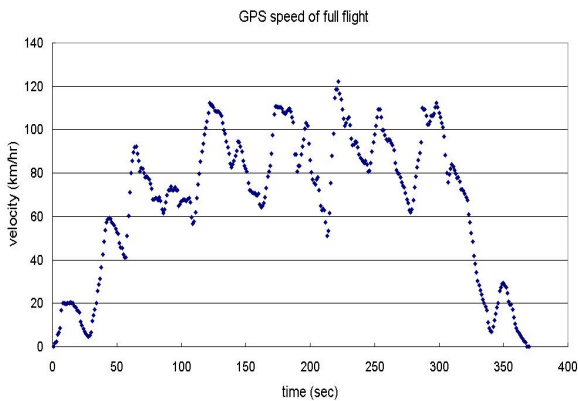

Figure 19 GPS altitude of the 2nd flight

Figure 20 GPS speed of the 2nd flight

Figure 21 shows a picture taken by the automatic camera on the UAV. The computer sends the command to trigger the camera payload module for automatically taking the pictures without the control from the base on the ground.
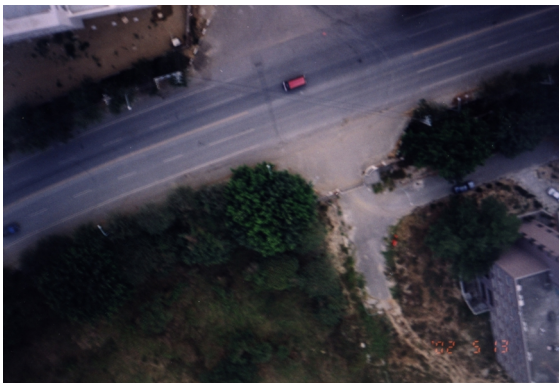

Figure 21 Aerial photo taken from UAV

7 Concluding Remark

The results of the present study of this project have shown that the functions of the onboard system and the ground station are verified both in the ground test and the flight test. Some significant features from this study can be conducted as following:

1. The onboard system has been developed successfully to have the capability of collecting data such as position, airspeed, altitude, heading, and health status of the UAV. The information allows for the implementation of the basic ability for the autonomous flight.
2. The features of the onboard software enable us to add any function without difficulty. It also shows the high efficiency and robust performance running on a simple industrial computer.
3. The success of the flight test reveals that the onboard system can operate normally and has overcome the environment of vibration and EMI problems in the UAV.
4. The success of data link for ground station has proven its value in flight tests to monitor real time UAV in-flight conditions and record flight data for further analysis.

8 Acknowledgement

References

[1] Hennessey, C., "*Autonomous Control of a Scale Airplane*", Simon Fraser University, 2000
[2] http://www.micropilot.com/
[3] Christophersen, H. B., Dhingra, M., Guily, R., Hart, M. G., Johnson, E. N., and Kahn, A., "*Development of an autonomous aerial reconnaissance system at Georgia Tech*", Georgia Institute of Technology, 2001
[4] Hsiao, F. B. and Lee, M. T., "The Development of Unmanned Aerial Vehicle in RMRL/NCKU," *4th Pacific International Conference on Aerospace Science and Technology*, Kaohsiung, Taiwan, May 2001.
[5] http://www.garmin.com/
[6] http://www.freewave.com/
[7] Lin, Y. R., "*The Development of an Onboard Computer System for Unmanned Aerial Vehicle*", Institute of Aeronautics and Astronautics, National Cheng Kung University, Master thesis, July 2000
[8] Jan Roskam, "*Airplane Design, Part VII* ", Roskam Aviation and Engineering Corporation, 1985
[9] Nelson, R. C., "*Flight stability and automatic control*" 2nd edition, McGraw-Hill Co.,1998
[10] Johnson, G. W., "*LabVIEW power programming*", McGraw-Hill, New York, ISBN 0079136664, 1998