بسم الله الرحمن الرحيم

Simulation of Aircraft Flight from Recorded Flight Data

A thesis submitted in partial fulfillment of the requirements for the degree of

M.Sc. Computer Engineering

Student

Elwaleed Mohammed Abdelwahid

Supervisor

Dr. Abdelrasoul Gabar Alzubaidy

DEPARTMENT OF ELECTRONIC ENGINEERING

COLLEGE OF ENGINEERING

SUDAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

July 2014

بسم الله الرحمن الرحيم

الله لا إله إلا هو الحي القيوم

صدق الله العظيم

DEDICATION

I would like to dedicate this thesis to my father,

who taught me that the best kind of knowledge to have is

that which is learned for its own sake.

I would like also to dedicate this thesis to my mother,

who taught me that even the largest task can be

accomplished if it is done one step at a time.

Lastly, I would like to dedicate this thesis to my inspire daughters …..

Al-zahra and Yasmeen.

ACKNOWLEDGEMENTS

ABSTRACT

The common flight data recorder (Тестер УЗ-ЛК) is the recorder of many Soviet Union aircraft types operated in Sudan, the use of the expensive Dvina-M software produced with this type of recorder for data analysis is fraught with major limitations and inabilities. These limitations affect pilot's training and force the countries operating these aircrafts to send their pilots to the aircrafts manufacturers for training. The main objective of this thesis is to produce software that simulates the flight session and produces the flight trajectory from a recorded flight data in the flight data recorder (FDR) so as to make the understanding of flight data analysis obvious and easy to discuss with students pilots.

Therefore, to simulate a flight session and reproduces its flight path from recorded flight parameters in flight data recorder (FDR) needs to classify, identify and extract the parameters that affect the movement of the aircraft. Then the selected parameters are reformed to meet the specifications of the Microsoft Flight Simulator parameters in order to pass those flight parameters through the FSUIPC to the MS Flight Simulator to simulate the aircraft behavior and motion during the real flight session.

The most important result of this research is the Real Flight Simulation software that can simulate and analyse pilot's flight training sessions, to reduce the effort, time and cost, with the ability to raise the capabilities and skills of pilot's competencies.

مـستـخـلـص

يعتبر مسجل بيانات الطلعات الجوية من النوع (Tectep У3-ЛК) هو المسجلالأكثر استخداماً في العديد من أنواع طائرات الاتحاد السوفيتي العاملة في السودان، ولكن استخدام البرمجيات المكلفة من النوع (Dvina-M) المنتجة مع هذا النوع من الأجهزة لتحليل البيانات، محفوف بقيود كثيرة وعجز كبير في طرق تحليل البيانات. هذه القيود تؤثر علي تدريب الطيارين مما يؤدي إلى إجبار الدول المشغلة لهذه الطائرات لإرسال طياريها للشركات المصنعة للطائرات لأغراض التدريب.

الهدف الرئيسي لهذه الأطروحة هو إنتاج البرامج التي تحاكي رحلة الطيران وتنتج مسار الرحلة من بيانات الرحلة المسجلة في مسجل بيانات الطلعات الجوية (FDR)، مما يجعل نتائج تحليل بيانات رحلة الطيران واضحة وسهلة المناقشة والفهم من قبل الطلاب الطيارين.

وعليه ولمحاكاة رحلة الطيران وإعادة استخراج مسار الرحلة من معلومات رحلة الطيران المسجلة في مسجل بيانات الطلعات الجوية (FDR) يحتاج إلى تصنيف وتحديد واستخراج البارامترات التي تؤثر على حركة الطائرات، ومن ثم يتم تهيئة البارامترات المختارة لتلبية مواصفات بارامترات برنامج محاكاة الطيران من شركة مايكروسوفت (MS FS)وذلك استعداداً لتمرير قيم تلك البارامترات من خلال الـ FSUIPC إلى MS FS مما ينتج عنه محاكاة لسلوك الطائرات وحركتها خلال رحلة الطيران الحقيقية.

ولذلك كانت أهم نتيجة لهذا البحث هو بناء البرنامج Real Flight Simulation الذي تمكن من محاكاة وتحليل طلعات تدريب الطيارين على الطيران وذلك لتقليل الجهد والوقت والتكلفة، مع القدرة على رفع قدرات ومهارات وكفاءات الطيارين.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

Chapter One

Introduction

1.1. Background

Data recorders on aircraft record informations about hundreds or thousands of aircraft operational parameters such as ground speed, pitch and altitude at a rate of multiple times per second.

Flight Data Analysis or Management of Flight Data is a process of translating the raw data of specific aircraft performance parameters that is recorded during the flight by Flight Data Recorder (FDR) into informations that expert engineers, pilots and accident investigators can use them to ensure highest standards of flight safety and the efficiency of the aircraft operations are maintained or to determine the causes of accidents, as well as assessing the performance of the trainee pilots.

This Analysis process usually done by special Softwares that used to convert the raw flight parameters data into a suitable form which assists the experts to extract accurate informations on the status of the aircraft during different stages of the flight.

Pilot training is one of the major fields of using these analysis softwares, the instructors pilot uses these softwares on the "After flight briefing session" to monitor, evaluate and correct the trainees pilots behavior during the flight training session. The great usefulness of these softwares appears when conducting pilots training on large aircrafts and fighter jets with high risk and operating cost which reduces the number of training hours that are required to get the trainee pilot's license relevant.

1.2. Problem Statement

Many of the aircrafts operating in Sudan and reliable in many tasks, use a flight data recorder with a very expensive flight data analysis software developed by the aircraft manufacturer. This software produces the results of the analysis only in form of tables and charts show the values of various

parameters during flight, these results are used most often by engineers only for the purposes of monitoring the performance of aircraft systems. However, this software does not show the flight path and model of the aircraft during flight, which represent the most important results of the analysis required by pilots for training purposes. Tables and charts are ambiguous and too boring for pilot's instructors to assess the students' performance during flying sessions. This palace in the software represents a major obstacle to the use of this software to reduce training costs and increase the efficiency of trainees during dangerous tasks that are assigned by.

1.3. Objectives

The main objective of this thesis is to produce software that simulates the flight session and produces the flight trajectory from a recorded flight data in the flight data recorder (FDR). The software will be used by pilot's instructors to discuss the training flight with the flight crew so as to save a great deal of time, thus it will make the understanding of flight data analysis obvious and easy to discuss with students pilots.

The software will ease the pain of aviation study for student pilots, by easing the understanding of the relationship between the various flying parameters and the movement of aircraft in the air, and the aircraft response speed to the use of control channels.

1.4. Methodology

The simulation of flight session from the recorded flight data requires gathering relevant data from specified documents and complete understanding of aircraft basic components and their relation to aerodynamics and theory of flight, the flight controls and how the airplane changes its flight attitude or position in flight. On the other hand as long as, pilot's role depend upon the instruments which give the pilot the sense and the necessary informations of the aircraft motion. Therefore, the

instruments used in aircraft motion sensing will be covered also, which will be the entrance to understand the flight parameters that affect the aircraft motion which are measured by instruments. Those parameters will be recorded by the automatic flight recording system and can be extracted in many data structure forms. Although the number of parameters are estimated in hundreds, and sometimes more than a thousand, but that requires the study of the recording device type and the method of data extraction and analysis, so that the parameters will be assembled, classified and tabulated. After that, the parameters that affect the movement of the aircraft and the flight track will be identified and extracted. Deep study of those parameters will be conducted so as to enable their use in simulations of aircraft movement and redraw its flight track.

The simulation phase begins with the study phase of the Flight Simulator, i.e its properties and methods of access to its own parameters for the purpose of reading and writing them, up to the mapping of the program and typing the code and testing it and ensuring the accuracy of the extracted path.

Finally, a comprehensive scientific project report will be written and it will be a subject of audition and correction as a preparation for susceptible to evaluation committees.

1.5. Project plan

The plan of this project is designed to get the overall benefits of the project and to achieve its objectives. The following timing diagram illustrates the plan for the project:

	Tasks	Months											
		1	2	3	4	5	6	7	8	9	10	11	12
1	Gathering and study the information and reference of Aircraft aerodynamics, aircraft controls and cockpit basic instruments.	■	■	■									
2	Get a clear description of parameters that affect the aircraft motion.		■	■									
3	Study of automatic flight data recording system and its design, configuration, operation and specifications.				■	■							
4	Gathering the specification and description of the Sudanese aircraft flight data recorder (Tectep y3).						■						
5	Master the process of extracting information from (Tectep y3) and get samples of flights data.							■					
6	Extract the parameters' data will be used in simulation of the flights.								■				
7	Prepare the Microsoft Flight Simulator with the modules and aircraft model for simulation.									■			
8	Master the FSUIPC utility and its memory addresses offsets, so as to communicate with the Flight Simulator.										■	■	
9	Write the simulation software Visual Basic code to communicate the recorded flight data with the Microsoft Flight Simulator.										■	■	
10	Software testing and modifying.											■	■
11	Editing the project report .	■	■	■	■	■	■	■	■	■	■	■	■

Figure (1.1): *Project plan timing diagram*

Chapter Two
Literature Review

2.1. Introduction

The flight path and behavior of an aircraft are determined by the interaction between the characteristics of the aircraft itself and the environment in which it is flying. The aircraft characteristics can be categorized as the physical characteristics, such as the shape, mass, volume, and surface area; the characteristics of the sub- systems, such as the propulsion, guidance, and control subsystems; and the structural characteristics, such as the loading and temperature limitations and the stiffness or rigidity of the structure.

The interaction of these quantities is the motion of the aircraft which sensed and measured by the instruments in the pilot's compartment. Those instruments give the pilot the sense and the necessary information of the aircraft motion.

As a mandatory requirement, the parameters measured by instruments recorded in an automatic flight recording system. This recorded information will be used when any unusual incidents or accident occurs for analysis and investigation activities.

Far-off accident investigation, the recorded data can be used to reform the flight session and flight path to monitor pilot actions in case of pilot training, and monitoring of aircraft operation performance status and technical condition.

2.2. Theory of flight and basic Components of Airplane

Figure (2.1) shows the main components of a high-winged airplane. The airframe consists of the *fuselage*, which is the main component of the airplane, the *wings*, and the *empennage*. The empennage (sometimes called the *tail feathers*) is the tail assembly consisting of the *horizontal stabilizer*, the *elevators*, the *vertical stabilizer*, and the *rudder*. The

elevators are used to adjust, or control, the pitch (nose up/down attitude) of the airplane. On some airplanes the entire horizontal stabilizer is the elevator, as shown in Figure (2.2). This is called a stabilator. The rudder is used to make small directional changes and in turns.

Most airplanes have small hinged sections on the trailing edge of the elevators and sometimes on the rudder called trim tabs as shown in Figure (2.2). These tabs move in the opposite direction to the control surface.
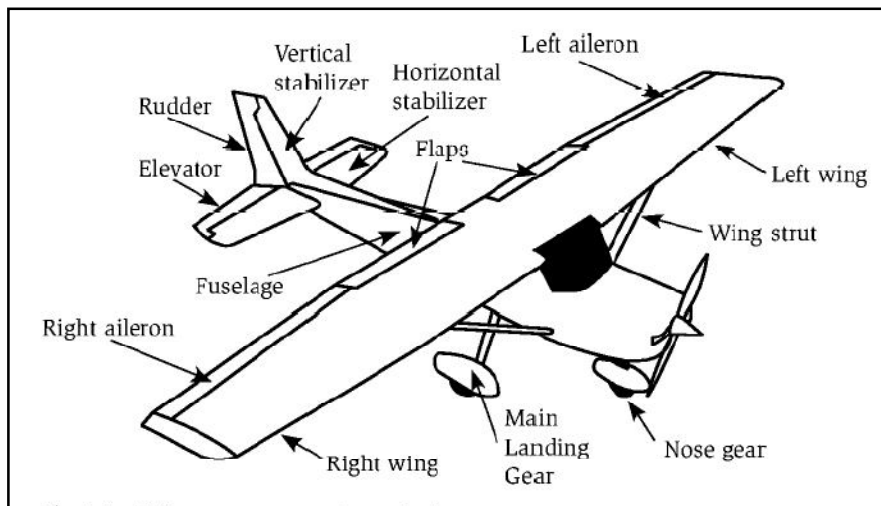
Figure (2.1): *Main component of an airplane*
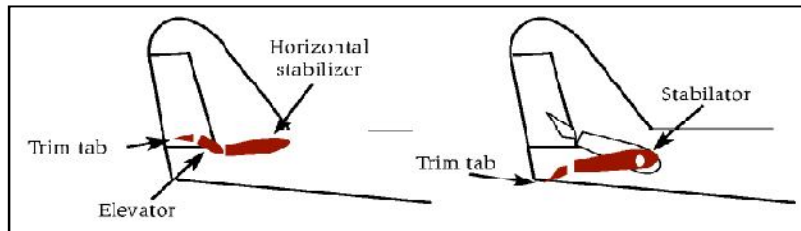
Figure (2.2): *The Empennage*

The purpose of the trim tabs is to reduce the necessary force on the control wheel, called a *yoke,* for the pilot to maintain a desired flight attitude.

The movable surfaces on the outer trailing edge of the wings are the *ailerons,* which are used for roll control (rotation around the axes of the fuselage). The hinged portions on the inboard part of the tailing edge of the wings are the *flaps*. These are used to produce greater lift at low speeds

and to provide increased drag on landing. This increased drag helps to reduce the speed of the airplane and to steepen the landing approach angle. Small airplanes have two configurations of landing gear. *Tricycle Landing gear* has the main landing gear just behind the center of balance of the airplane and a steerable nose gear up forward. The *tail dragger* has the main landing gear forward of the center of balance and a small steerable wheel at the tail. The nose gear and the tail wheel are steered with the rudder pedals.

2.2.1. Airfoils and Wings

An *airfoil* is a shape designed to produce lift. As shown in Figure 1.3, an airfoil is the shape seen in a slice of a wine. Beside the wine, propellers and the tail surfaces are also airfoils. For some wings, slices taken at different places along its length will reveal different airfoils.

An airfoil has a leading *edge* and a *trailing edge.* As detailed in Figure (2.4) a *chord* and a *camber* also characterize an airfoil. The chord is an imaginary straight line connecting the leading edge with the trailing edge. The chord is used for determining the geometric *angle of attack* and for determining the area of a wing.

The *mean camber line* is the line an equal distance from the upper and lower surfaces of the wing. The camber is the curvature of the mean camber line. A wing that has an airfoil with a great deal of curvature in its mean camber line is said to be a highly cambered wing. A symmetric airfoil has no camber.
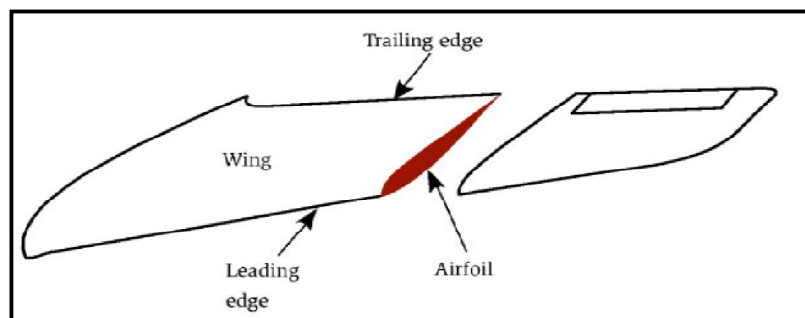
Figure (2.3): *The Wing and airfoil*
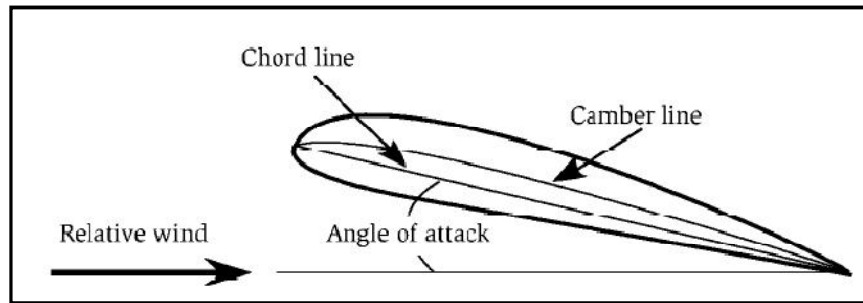
Figure (2.4): *Airfoil nomenclature*

An airfoil with lift also has an angle of attack, as shown in the Figure (2.4), the *relative wind is* the direction of the wind at some distance from the wing. It is parallel to and opposite to the direction of motion of the wing. The velocity of the relative wind is equal to the speed of the wing. In aeronautics the *geometric angle of attack* is defined as the angle between the *mean chord* of the airfoil and the direction of the relative wind.

A useful measure of a wing is its aspect ratio. The aspect ratio is defined as the wing's *span* divided by the average or *mean chord length*. The span is the length of the wing measured from wingtip to wingtip. The mean chord length is the average chord length along the wing. The area of the wing is just the span times the mean chord length.

2.2.2. The Four Forces

There are four forces associated with the flight of an airplane. These forces, illustrated in Figure (2.5), are lift, weight, thrust, and drag. In straight-and-level flight (not changing speed, direction, or altitude) the net lift on the airplane is equal to its weight. The thrust produced by the engine is equal to the drag, which is caused by air friction and the work done produce the lift.
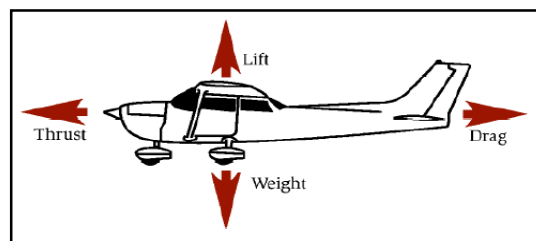
Figure (2.5): *The four forces on an airplane in straight-and-level flight*

2.2.3. Theory of Flight

The physical description of lift is useful for an accurate understanding of flight. This description demonstrates that aircraft flight based primarily on Newton's laws and Bernoulli's theorem.

2.2.3.1. Physical description of lift

Referring to Newton's first law to change the aircraft status from rest to flight and to generate lift a wing must do something to the air. What the wing does to the air is the action while lift is the reaction(Newton's 3rd law) In Figure (2.6) the air comes from below the wing (this is the upwash) splits and passes over the wing and is bent down. Newton's first law says that there must be a force on the air to bend it down (the action). Newton's third law says that there must be an equal and opposite force (up) on the wing (the reaction). To generate lift a wing must divert lots of air down.
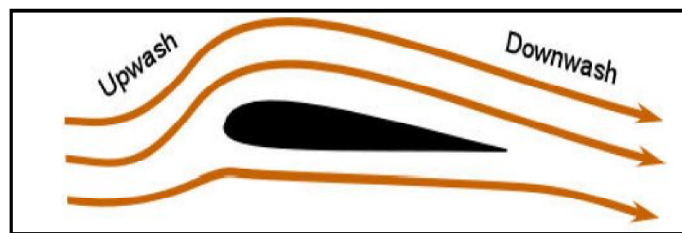
Figure (2.6): *True airflow of a wing with lift*

The lift of a wing is equal to the change in momentum of the air it is diverting down. Momentum is the product of mass and velocity (mv). The most common form of Newton's second law is (F= ma), or force equal mass times acceleration. The law in this form gives the force necessary to accelerate an object of a certain mass. An alternate form of Newton's second law can be written:

The lift of a wing is proportional to the amount of air diverted down times the vertical velocity of that air.

Thus, for more lift the wing can either divert more air (mass), increase its vertical velocity or a combination of the two. This vertical velocity behind the wing is the vertical component of the "downwash".

When the air is bent around the top of the wing, it pulls on the air above it accelerating that air downward. Otherwise there would be voids in the air above the wing. Air is pulled from above. This pulling causes the pressure to become lower above the wing. It is the acceleration of the air above the wing in the downward direction that gives lift.

In Figure (2.7), to bend the air requires a force. As indicated by the colored arrows, the direction of the force on the air is perpendicular to the bend in the air. The magnitude of the force is proportional to the tightness of the bend. The tighter the air bends the greater the force on it. The forces on the wing, as shown by the black arrows in the figure, have the same magnitude as the forces on the air but in the opposite direction. These forces, working through pressure, represent the mechanism in which the force is transferred to the wing.

Force on wing Force on air
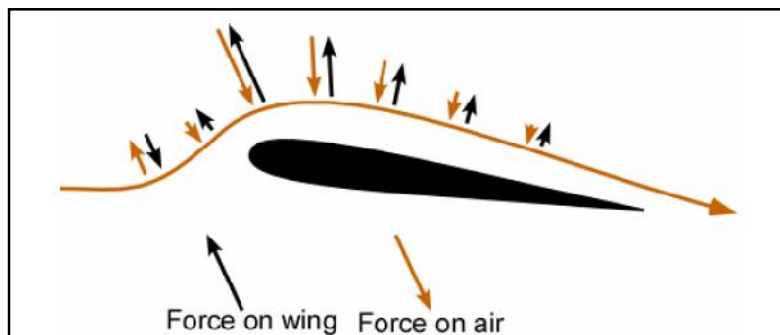
Figure (2.7): *Forces on the air and the corresponding reaction forces on the wing*

Thus, in brief, the air bends around the wing producing downwash. Newton's first law says that the bending of the air requires a force on the air, and Newton's third law says that there is an equal and opposite force on the wing. That is a description of lift. The pressure difference across the wing is the mechanism in which lift is transferred to the wing due to the bending of the air.

2.2.3.2. Lift requires power

When a plane passes overhead the formally still air gains a downward velocity. Thus, the air is left in motion after the plane leaves. The air has been given energy. Power is energy, or work, per time. So, lift requires power. This power is supplied by the airplane's engine.

We have already stated that the lift of a wing is proportional to the amount of air diverted times the vertical velocity of that air. Thus, the power needed to lift the airplane is proportional to the load (or weight) times the vertical velocity of the air. If the speed of the plane is doubled, the amount of air diverted down also doubles. Thus to maintain a constant lift, the angle of attack must be reduced to give a vertical velocity that is half the original. The power required for lift has been cut in half. This shows that the power required for lift becomes less as the airplane's speed increases. In fact, this power to create lift is proportional to 1/speed of the plane.

But, there must be more power than the power required for lift. The power associated with lift is often called the "induced" power. Power is also needed to overcome what is called "parasite" drag, which is the drag associated with moving the wheels, struts, antenna, etc. through the air. The energy the airplane imparts to an air molecule on impact is proportional to the speed^2 (Kinetic Energy=½(mass)×(velocity)^2 or ½mv^2). The number of molecules struck per time is proportional to the speed. The faster one goes the higher the rate of impacts. Thus the parasite power required to overcome parasite drag increases as the speed^3. Figure (2.8) shows the "power curves" for induced power, parasite power, and total power (the sum of induced power and parasite power). Again, the induced power goes as 1/speed and the parasite power goes as the speed^3.
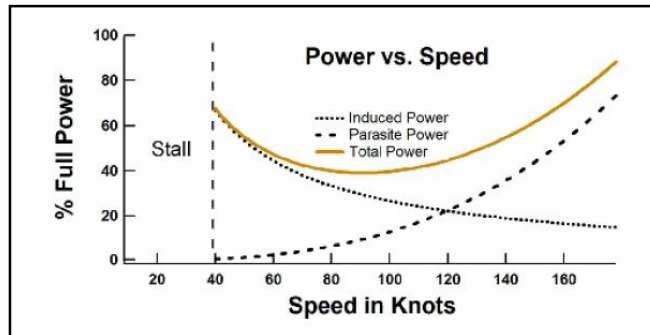
Figure (2.8): *Power vs. Speed*

2.2.4. Flight controls

Whenever an airplane changes its flight attitude or position in flight, it rotates about one or more of three axes, which are imaginary lines that pass through the airplane's center of gravity each is at a 90° angle to the other two. The three axes are longitudinal, lateral and vertical axes, which shown in Figure (2.9) with the airplane motions about each.

The three motions of the airplane (roll, pitch, and yaw) are controlled by three control surfaces. Roll is controlled by the ailerons; pitch is controlled by the elevators; yaw is controlled by the rudder.

Aircraft flight control systems are classified as primary and secondary. The primary control systems consist of those that are required to safely control an airplane during flight. These include the ailerons, elevator (or stabilator), and rudder. Secondary control systems improve the performance characteristics of the airplane, or relieve the pilot of excessive control forces. Examples of secondary control systems are wing flaps and trim systems.
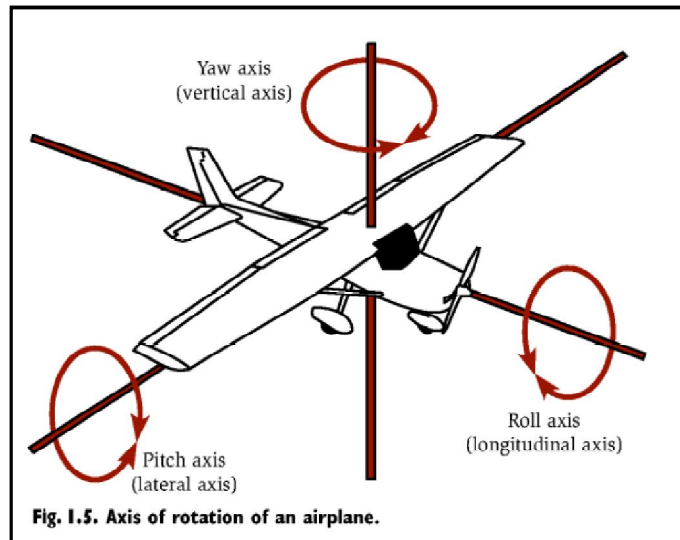
Figure (2.9): *Axes of an airplane.*

2.2.4.1. Primary flight controls

Airplane control systems are carefully designed to provide a natural feel, and at the same time, allow adequate responsiveness to control inputs. At low airspeeds, the controls usually feel soft and sluggish, and the airplane responds slowly to control applications. At high speeds, the controls feel firm and the response is more rapid. Movement of any of the three primary flight control surfaces changes the airflow and pressure distribution over and around the airfoil. These changes affect the lift and drag produced by the airfoil and control surface combination, and allow a pilot to control the airplane about its three axes of rotation.

Design features limit the amount of deflection of flight control surfaces. The purpose of these design limits is to prevent the pilot from inadvertently over controlling and overstressing the aircraft during normal maneuvers.

2.2.4.1.1. Ailerons

Ailerons control roll about the longitudinal axis. The ailerons are attached to the outboard trailing edge of each wing and move in the opposite direction from each other. Ailerons are connected by cables, bell cranks, pulleys or push-pull tubes to each other and to the control wheel.

Moving the control wheel to the right causes the right aileron to deflect upward and the left aileron to deflect downward. The upward deflection of the right aileron decreases the camber resulting in decreased lift on the right wing. The corresponding downward deflection of the left aileron increases the camber resulting in increased lift on the left wing. Thus, the increased lift on the left wing and the decreased lift on the right wing cause the airplane to roll to the right.
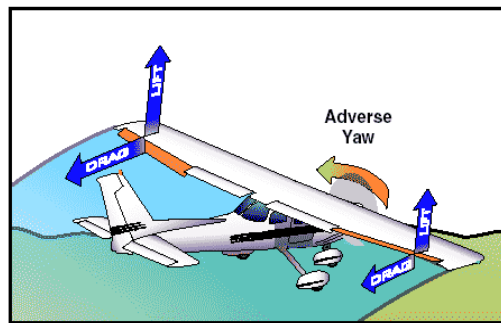
Figure (2.10)*: Adverse yaw.*

Since the downward deflected aileron produces more lift, it also produces more drag. This added drag attempts to 'yaw' the airplane's nose in the direction of the raised wing. This is called adverse yaw.

The rudder is used to counteract adverse yaw, and the amount of rudder control required is greatest at low airspeeds, high angles of attack, and with large aileron deflections. However, with lower airspeeds, the vertical stabilizer and rudder combination becomes less effective, and magnifies the control problems associated with adverse yaw.

During a turn, the angle of attack must be increased by applying elevator pressure because more lift is required than when in straight-and level flight. The steeper the turn, the more back elevator pressure is needed.

As the desired angle of bank is established, aileron and rudder pressures should be relaxed to stop the bank from increasing because the aileron and rudder control surfaces will be neutral in their streamlined position. Elevator pressure should be held constant to maintain a constant altitude.

The rollout from a turn is similar to the roll-in except the flight controls are applied in the opposite direction toward the high wing. As the angle of bank decreases, the elevator pressure should be relaxed as necessary to maintain altitude.

With differential ailerons, one aileron is raised a greater distance than the other aileron is lowered for a given movement of the control wheel. This produces an increase in drag on the descending wing. The greater drag results from deflecting the up aileron on the descending wing to a greater angle than the down aileron on the rising wing. While adverse yaw is reduced, it is not eliminated completely.

With a frise-type aileron (Figure 2.11), when pressure is applied to the control wheel, the aileron that is being raised pivots on an offset hinge. This projects the leading edge of the aileron into the airflow and creates drag. This helps equalize the drag created by the lowered aileron on the opposite wing and reduces adverse yaw.
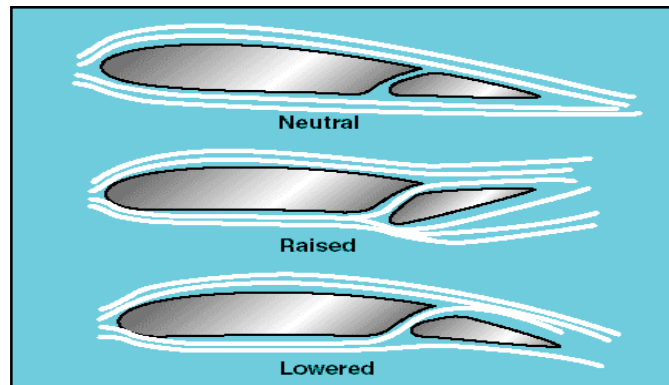
Figure (2.11): *Frise-type ailerons.*

The Frise-type aileron also forms a slot so that air flows smoothly over the lowered aileron, making it more effective at high angles of attack. Frise-type ailerons also may be designed to function differentially. Frise-type aileron does not eliminate adverse yaw entirely. Coordinated rudder application is still needed wherever ailerons are applied.

2.2.4.1.2. Rudder

The rudder controls movement of the airplane about its vertical axis. This motion is called yaw. Like the other primary control surfaces, the rudder is a movable surface hinged to a fixed surface, in this case, to the vertical stabilizer, or fin. Moving the left or right rudder pedal controls the rudder.

When the rudder is deflected into the airflow, a horizontal force is exerted in the opposite direction. By pushing the left pedal, the rudder moves left. This alters the airflow around the vertical stabilizer and rudder, and creates a sideward lift that moves the tail to the right and yaws the nose of the airplane to the left.

Rudder effectiveness increases with speed, so large deflections at low speeds and small deflections at high speeds may be required to provide the desired reaction.
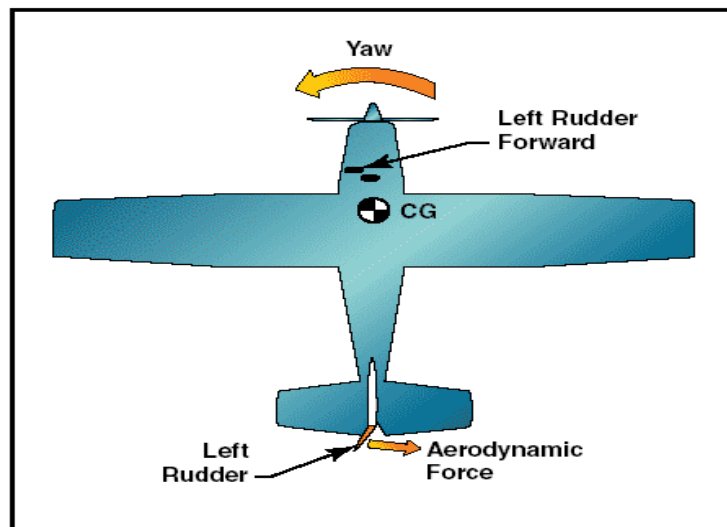
Figure (2.12): *The effect of left rudder pressure.*

2.2.4.1.3. Elevator

The elevator controls pitch about the lateral axis. Like the ailerons on small airplanes, the elevator is connected to the control column in the cockpit by a series of mechanical linkages. Aft movement of the control column deflects the trailing edge of the elevator surface up. This is usually

referred to as up elevator. The up-elevator position decreases the camber of the elevator and creates a downward aerodynamic force, which is greater than the normal tail-down force that exists in straight-and-level flight. The overall effect causes the tail of the airplane to move down and the nose to pitch up. The pitching moment occurs about the center of gravity (CG).
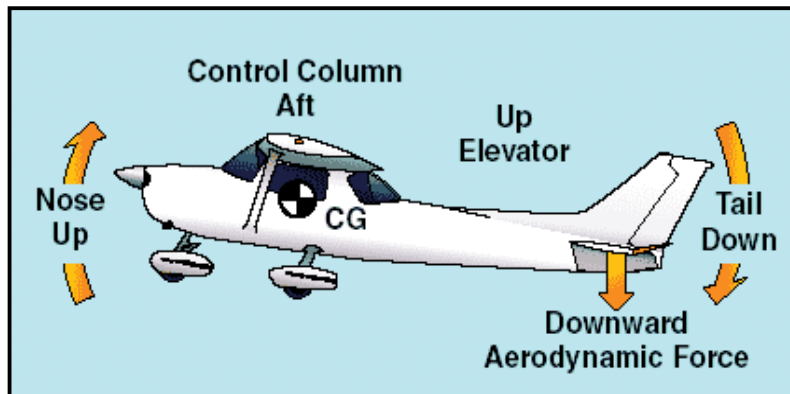
Figure (2.13): *The elevator used for changing the pitch attitude*

The strength of the pitching moment is determined by the distance between the CG and the horizontal tail surface, as well as by the aerodynamic effectiveness of the horizontal tail surface.

Moving the control column forward has the opposite effect, elevator camber increases, creating more lift (less tail-down force) on the horizontal stabilizer/elevator. This moves the tail upward and pitches the nose down. Again, the pitching moment occurs about the CG.

In a T-tail configuration, the elevator is above most of the effects of downwash from the propeller as well as airflow around the fuselage and/or wings during normal flight conditions. Operation of the elevators in this undisturbed air makes control movements consistent throughout most of flight regimes. T-tail designs have become popular on many light and large airplanes, especially those with aft-fuselage mounted engines, because the T-tail configuration removes the tail from the exhaust blast of the engines. Seaplanes and amphibians often have T-tails in order to keep the horizontal surfaces as far from the water as possible. An additional benefit is reduced vibration and noise inside the aircraft.

At slow speeds, the elevator on a T-tail aircraft must be moved through a larger number of degrees of travel to raise the nose a given amount as compared to a conventional-tail aircraft. This is because the conventional-tail aircraft has the downwash from the propeller pushing down on the tail to assist in raising the nose.

When flying at a very high angle of attack with a low airspeed and an aft CG, the T-tail airplane may be susceptible to a deep stall (Figure 1.14). In a deep stall, the airflow over the horizontal tail is blanketed by the disturbed airflow from the wings and fuselage. In these circumstances, elevator or stabilator control could be diminished, making it difficult to recover from the stall.
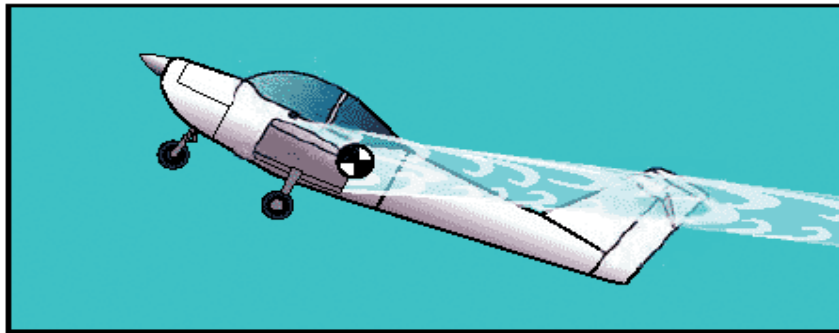
Figure (2.14): *A T-tail at a high angle of attack and an aft CG.*

Many airplanes have systems to compensate for this situation. The systems range from "elevator down spring" which assists in lowering the nose to prevent a stall caused by the aft CG position (Figure 1.15).
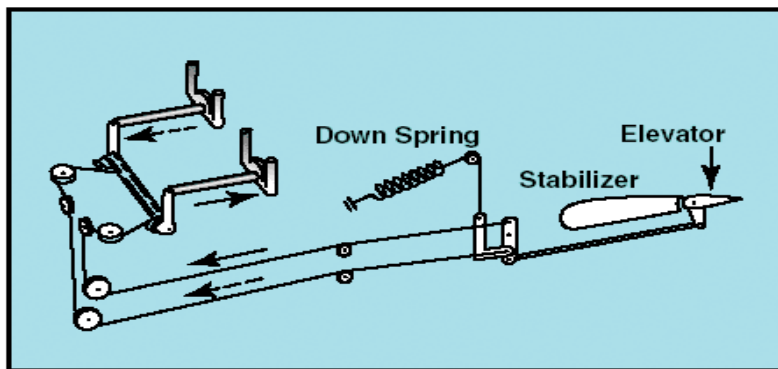
Figure (2.15): *Elevator down spring*

Stabilator (Figure 1.16) is essentially a one-piece horizontal stabilizer with the same type of control system. Because stabilator pivot around a central hinge point, it is extremely sensitive to control inputs and aerodynamic loads. Antiservo tabs are incorporated on the trailing edge to decrease sensitivity in order to prevent over-controlling from pilot-induced control inputs. A balance weight is usually incorporated ahead of the main spar.
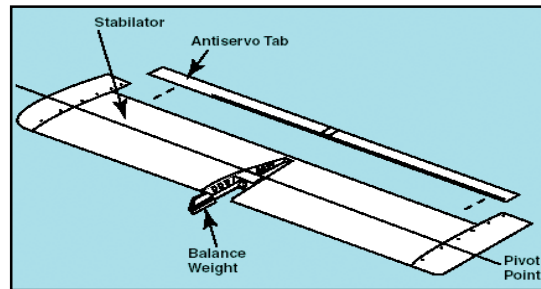
Figure (2.16): *The stabilator is a one-piece horizontal tail surface*

2.2.4.2. Secondary flight controls

Main secondary flight control systems consist of the flaps, leading edge devices, and trim devices.

2.2.4.2.1. Flaps

Flaps are the most common high-lift devices attached to the trailing edge of the wing increase both lift and induced drag for any given angle of attack. Figure (2.17) shows the common types of flaps.
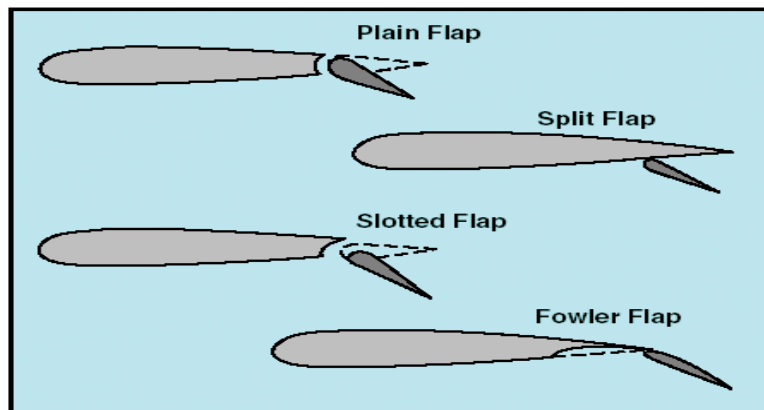
Figure (2.17): *Four common types of flaps.*

The plain flap is the simplest of the four types. It increases the airfoil camber, resulting in a significant increase in the coefficient of lift at a

given angle of attack. It greatly increases drag and moves the center of pressure aft on the airfoil, resulting in a nose-down pitching moment.

The split flap is deflected from the lower surface of the airfoil and produces a slightly greater increase in lift than does the plain flap. However, more drag is created because of the turbulent air pattern produced behind the airfoil. When fully extended, both plain and split flaps produce high drag with little additional lift.

The most popular flap on airplanes today is the slotted flap. It increases the lift coefficient significantly more than plain or spilt flaps. When the slotted flap is lowered, high-energy air from the lower surface is ducted to the flap's upper surface. The high-energy air from the slot accelerates the upper surface boundary layer and delays airflow separation, providing a higher coefficient of lift.

Fowler flaps are a type of slotted flap. This flap design not only changes the camber of the wing, it also increases the wing area. Instead of rotating down on a hinge, it slides backwards on tracks. In the first portion of its extension, it increases the drag very little, but increases the lift a great deal as it increases both the area and camber. As the extension continues, the flap deflects downward, and during the last portion of its travel, it increases the drag with little additional increase in lift.

2.2.4.2.2. Leading edge devices

High-lift devices can be applied to the leading edge of the airfoil. The most common types are fixed slots, movable slats, and leading edge flaps (Figure 1.18).
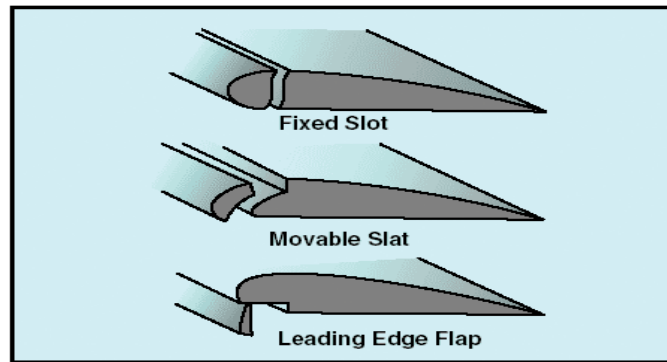
Figure (2.18): *Leading edge high lift devices.*

Fixed slots direct airflow to the upper wing surface and delay airflow separation at higher angles of attack. The slot does not increase the wing camber, but allows a higher maximum coefficient of lift because the stall is delayed until the wing reaches a greater angle of attack.

Movable slats consist of leading edge segments, which move on tracks. At low angles of attack, each slat is held flush against the wing's leading edge by the high pressure that forms at the wing's leading edge. As the angle of attack increases, the high-pressure area moves aft below the lower surface of the wing, allowing the slats to move forward. Some slats, however, are pilot operated and can be deployed at any angle of attack. Opening a slat allows the air below the wing to flow over the wing's upper surface, delaying airflow separation.

Leading edge flaps, like trailing edge flaps, are used to increase the camber of the wings. This type of leading edge device is frequently used in conjunction with trailing edge flaps and can reduce the nose-down pitching movement produced by the latter.

2.2.4.2.3. Trim systems

Trim systems are used to relieve the pilot of the need to maintain constant pressure on the flight controls. Trim systems usually consist of cockpit controls and small hinged devices attached to the trailing edge of one or more of the primary flight control surfaces. They are designed to assist movement and position of the flight control surface to which they are

attached. Common types of trim systems include trim tabs, balance tabs, antiservo tabs, ground adjustable tabs, and an adjustable stabilizer.

2.3. Flight instruments

Pilot's roles during flight are sensing of current motion of the aircraft, make decision and compute appropriate control signals for the next motion, and sensing response of aircraft and provide feedback as necessary. Therefore, the pilot's role depend upon the instruments which give the pilot the sense and the necessary information of the aircraft motion, and the flight controls which enables the pilot to react and give feedback to maintain the correct flight path.

Basically there are six flight instruments whose indications are so coordinated as to create a 'picture' of an aircraft's flight condition; they are, altimeter, airspeed indicator, vertical speed indicator, gyro horizon, , turn-and-bank indicator and direction indicator. In addition, for pilot training purposes we add engine speed indicator (RPM indicator). Usually, these instruments properly grouped in the instruments panel in the cockpit to maintain coordination and to assist a pilot to observe them with the minimum of effort.

2.3.1. Pitot-static instruments and systems

The pitot-static system of an aircraft is a system in which total pressure created by the forward motion of the aircraft and the static pressure of the atmosphere surrounding it, are sensed and measured in terms of speed, altitude and rate of change of altitude (vertical speed). In other words, the system may be referred to as a manometric, or air data system.

In its basic form the system consists of a pitot-static tube, or probe, the three primary flight instruments (airspeed, altimeter and vertical speed) and pipelines, interconnected as shown in Figure (2.19).

Sensing of the total, or pitot pressure, and the static pressure is effected by the probe, which is suitably located in the air stream and transmits these pressures to the instruments.
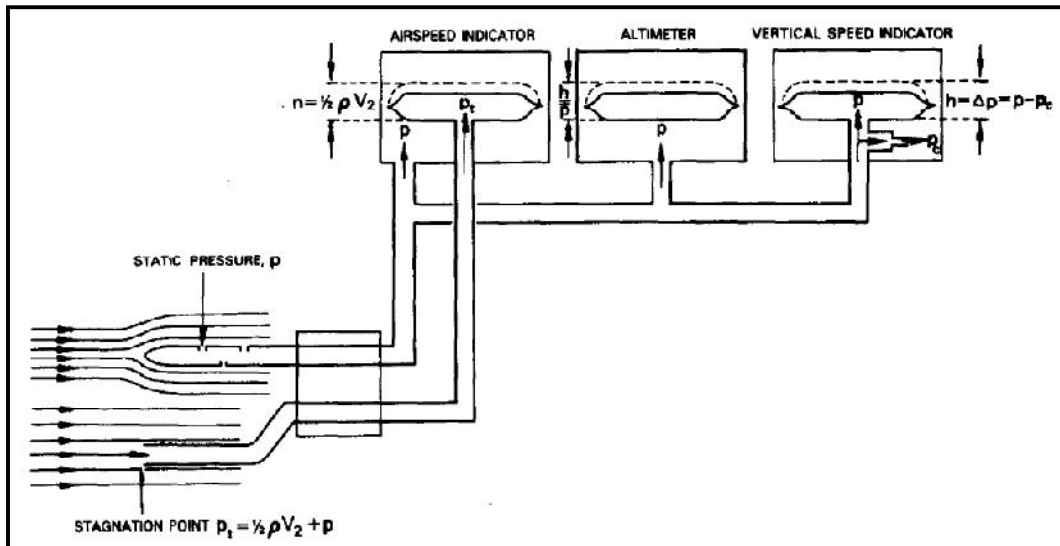
Figure (2.19): *Sensing and transmission of pitot and static pressures.*

In Figure (2.19) the probe is shown in its simplest form consists of two forward-facing tubes positioned parallel to each other and in a vertical plane. One of the tubes, the pitot tube, is open at its forward end to receive the total air pressure resulting from the aircraft's forward movement, while the other, the static tube, is closed at its forward end but has a series of small holes drilled circumferentially at a calculated distance from the forward end through which the undisturbed air at prevailing atmospheric pressure is admitted. Pressures are transmitted to the instruments through pipelines connected to each tube.

Pitot pressure defined as the additional pressure produced on a surface when a flowing fluid is brought to rest, or stagnation, at the surface. If a probe placed in a fluid with its open end facing upstream as shown in Figure (2.20) and the fluid flows at a certain velocity (*Y*) over the probe, it will be brought to rest at the nose and this point is known as the stagnation point. If the fluid is an ideal one (not viscous), then the total energy is equal to the sum of the potential energy, the kinetic energy and pressure

energy, and remains constant. In connection with a pitot probe however, the potential energy is neglected thus leaving the sum of the remaining two terms as the constant. Now, in coming to rest at the stagnation point, kinetic energy of the fluid is converted into pressure energy.
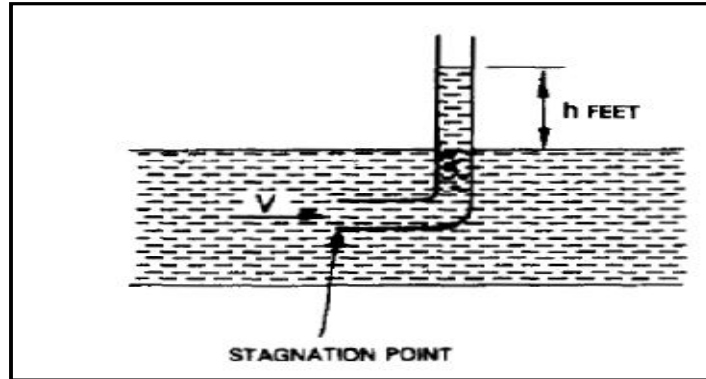
Figure (2.20): *Pitot pressure*

This means that work must be done by the mass of liquid and this raises an equal volume of the fluid above the level of the fluid stream. If the mass of the fluid above this level is (m) pounds then the work done in raising it through a height (h) foot is given by:

$$\text{Work done} = mgh \quad \text{foot-pounds}$$

Where (g) is the acceleration due to gravity.

The work done is also equal to the product of the ratio of mass to density (ρ) and pressure (p):

$$Work\ done = \frac{m}{\rho} \times p$$

The kinetic energy of a mass (m) before being brought to rest is equal to $\frac{1}{2}mV^2$, where V is the speed, and since this is converted into pressure

energy, $\frac{m}{\rho} \times p = \frac{1}{2}mV^2$

Therefore $p = \frac{1}{2}\rho V^2$

2.3.1.1. Measurement of Altitude

In the general development of instruments for atmospheric pressure measurement, certain practical applications demanded instruments which would be portable and able to operate in various attitudes.

Thus the aneroid barometer came into being. The instrument has been developed to quite a high standard of precision. The pressure-sensing element of the instrument (Figure 2.21) is an evacuated metal capsule. Since there is approximately zero pressure inside the capsule, and assuming the instrument to be at sea-level (approximately 14.7 lbf/in² on the outside) the capsule will tend to collapse. This, however, is prevented by a strong leaf spring fitted so that one side is attached to the top of the capsule and the other side to the instrument base plate. The spring always tends to open outwards, and a state of equilibrium is obtained when the pressure is balanced by the spring tension.

If now the atmospheric pressure decreases, the force tending to collapse the capsule is decreased but the spring tension remains the same and consequently is able to open out the capsule a little further than before. If there is an increase in pressure, the action is reversed, the pressure now collapsing the capsule against the tension of the spring until equilibrium is attained.
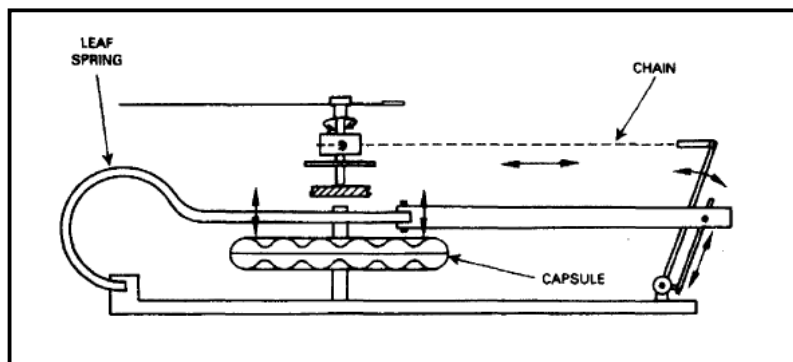
Figure (2.21): *Aneroid barometer.*

The resulting expansion and contraction of the capsule, which is extremely small, is transformed into rotary motion of the pointer by means of a magnifying lever system and a very finely-linked chain.

Present-day altimeters are, much more sophisticated, the pressure-sensing element is made up of three aneroid capsules stacked together to increase the sensitivity of the instrument. In the most notable altimeter, the triple-pointers type (Figure 2.22), deflections of the capsules are transmitted to a sector gear meshes with a magnifying gear mechanism which drives a hand staff carrying a long pointer the function of which is to indicate hundreds of feet. Second gear mechanism carrying second and third pointers which indicate thousands and tens of thousands of feet respectively. In this particular instrument a disc is also attached to the third pointer gear and moves with it.

Figure (2.22): *Triple-pointers altimeter*

In order, to compensate for altitude errors due to atmospheric pressure changes, altimeters are provided with a manually operated adjustment device which allows the pointers to be set to zero height for any prevailing ground pressure so that the indications in flight will still be heights in the standard atmosphere above the ground.

The adjustment device principle is shown in Figure (2.23), a scale-type device in very simple form. The scale is mounted on a gear which meshes with a pinion on the end of the control knob shaft and also with the pointer

gearing. A differential gear mechanism (not shown) allows the pointer to be rotated without disturbing the setting of the capsules.

Similarly, any setting of the altitude pointer automatically adjusts the millibar scale reading to indicate the pressure at which the height indicated will be zero.
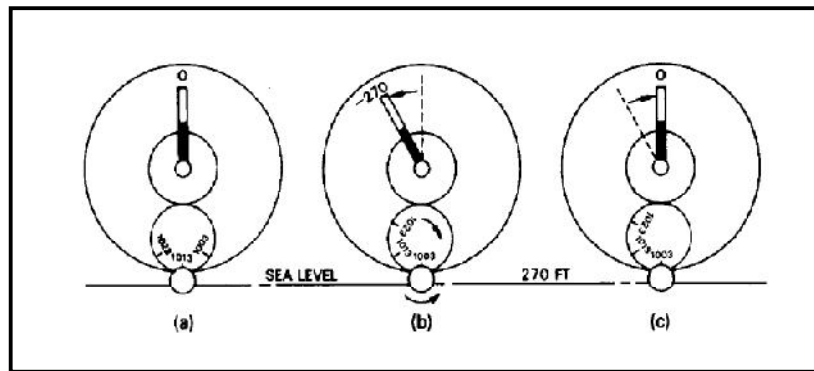
Figure (2.23): *Principle of barometric pressure adjustment*

2.3.1.2. Airspeed Indicators

Airspeed indicators are in effect very sensitive pressure gauges measuring the difference between the pitot and static pressures detected by the pressure probe, in terms of the $\frac{1}{2}\rho V^2$ formula. Originally this formula served as the basis for the calibration of indicators, but as airspeeds began to increase a noticeable error began to creep into the indications. The reason for this was that, as the high-velocity air was brought to 'stagnation' at the pitot tube, it was compressed and its density was increased. The formula was therefore modified to take into account additional factors in order to minimize the 'compressibility error'. Thus, present-day airspeed indicators are calibrated to the law:

$$p = \frac{1}{2}\rho V^2 \left(1 + \frac{V^2}{4a_0^2}\right)$$

Where p = Pressure difference (mm H_2O)

p = Density of air at sea-level.

V = Speed of aircraft (m.p.h. or knots)

a_0 = Speed of sound at sea-level (m.p,h.)

The numerical values to be inserted in the formula depend on whether (V) is expressed in m.p.h. or knots.

$$p = 0.012504\ V^2(1 + 0.43\ V^2 \times 10^{-16}) \text{ with V in m.p.h.}$$
$$= 0.016580\ V^2(1 + 0.57\ V^2 \times 10^{-16}) \text{ with V in knots.}$$

The mechanism of a typical simple-type airspeed indicator is illustrated in Figure (2.24). The pressure sensing element is a metal capsule, the interior of which is connected to the pitot pressure connector via a short length of capillary tube. Static pressure is exerted on the exterior of the capsule and is fed into the instrument case via the second connector. Except for this connector the case is sealed.
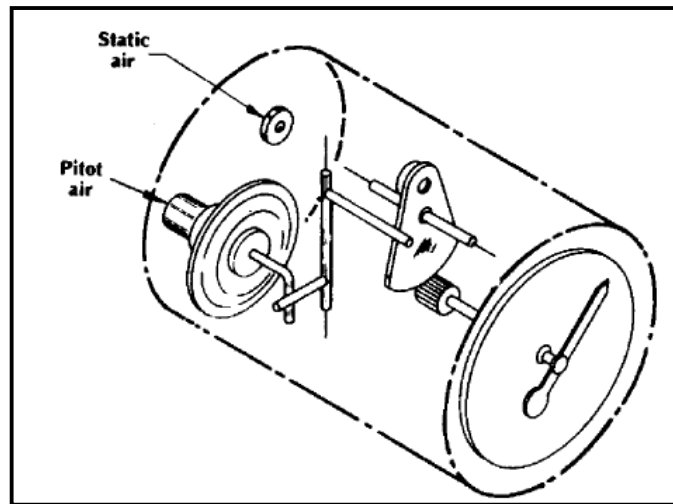
Figure (2.24): *A typical simple-type airspeed indicator*

Displacements of the capsule in accordance with what is called the 'square-law' are transmitted via a magnifying lever system, gearing, and a square-law compensating device to the pointer, which moves over a scale calibrated in m.p.h or knots. Temperature compensation is achieved by a bimetallic strip arranged to vary the magnification of the lever system in opposition to the effects of temperature on system and capsule sensitivity.

2.3.1.3. Machmeter

The forces which the aircraft structure experiences at high speed due to air resistance are dependent on how close the aircraft's speed approaches that

of sound. Since the speed of sound depends on atmospheric pressure and density, it will vary with altitude, and this suggests that in order to fly an aircraft within safe speed limits a different airspeed would have to be maintained for each altitude. Therefore, the ratio of the aircraft's speed (V), and the speed of sound (a), could be computed from pressure measurement and indicated in a conventional manner. This ratio, (V/a), is termed the Mach number (M), and the instrument which measures the ratio (M = V/a) is termed a *Machmeter.*

The Mach number at which the air resistance causing turbulence and buffeting of the aircraft, is referred to as the *Critical Mach number (M_{crit})* of that particular aircraft and being a ratio of air speed and sonic speed, it will be the same for any altitude and indicated by a pre-adjusted mark located over the dial of the Machmeter.

A Machmeter is a compound flight instrument which accepts two variables and uses them to compute the required ratio. The construction of the instrument is shown in Figure (2.25).

The first variable is airspeed and therefore a mechanism based on the conventional airspeed indicator is adopted to measure this in terms of the pressure difference (p–s), where (p) is the total or pitot pressure and (s) is the static pressure. The second variable is altitude, and this is also measured in the conventional manner, by means of an aneroid capsule sensitive to the static pressure (s). Deflections of the capsules of both mechanisms are transmitted to the indicator pointer by rocking shafts and levers, the dividing function of the altitude unit being accomplished by an intermediate sliding rocking shaft.
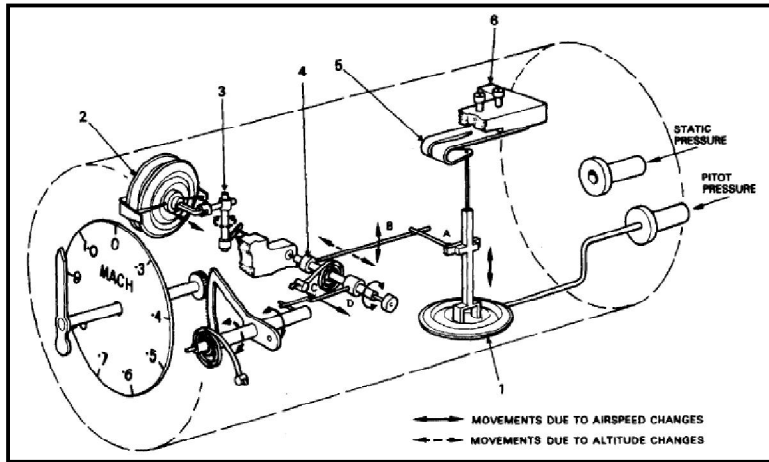
Figure (2.25): *Machmeter*

2.3.1.4. Maximum allowable airspeed/computed Airspeed indicator

This indicator as shown in Figure (2.26) looks much like a standard airspeed indicator, but has an additional striped pointer for the maximum airspeed actuated by an aneroid, or altimeter mechanism, that moves it to a lower value as air density decreases. This instrument, however, has the additional feature of indicating the airspeed computed by a central air data computer. The indicating element for the computed air speed (CAS) is a servo motor driven digital counter, the motor being supplied with signals from a synchronous transmission system.
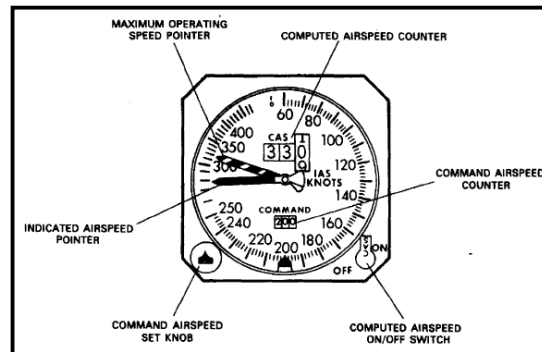
Figure (2.26): *Indicated/computed airspeed indicator*

2.3.1.5. Vertical Speed Indicator

These indicators, also known as rate-of-climb indicators, are the third of the primary group of pitot-static flight instruments, and are very sensitive

differential pressure gauges, designed to indicate the rate of altitude change from the change of static pressure alone.

The operative clause of this indicator is 'the rate at which the static pressure changes', and as this involves a time factor we have to introduce this into the measuring system as a pressure function. It is accomplished by using a special air metering unit, and it is this which establishes the second pressure required.

An indicator consists basically of three main components, a capsule, an indicating element and a metering unit, all of which are housed in a sealed case provided with a static pressure connection at the rear. The dial presentation is such that zero is at the 9 o'clock position; thus the pointer is horizontal during straight and level flight and can move from this position to indicate climb and descent in the correct sense.

The indicator mechanism is shown in schematic form in Figure (2.27), from which it will be noted that the metering unit forms part of the static pressure connection and is connected to the interior of the capsule by a length of capillary tube. The other end of the metering unit is open to the interior of the case to apply static pressure to the exterior of the capsule. The instrument operation under the three flight conditions: (a) level flight, *(b)* descent, and (c) climb are shown in schematic form in Figure (2.27).

The metering unit is a specially calibrated leak assembly designed to restrict the flow of air into or out of the instrument case approximately four seconds than the flow of air into or out of the capsule. Thus, the metering unit introduces the required rate and time-lag factors, and differential pressure across the capsule which positions the pointer to indicate the altitude change in feet per minute.
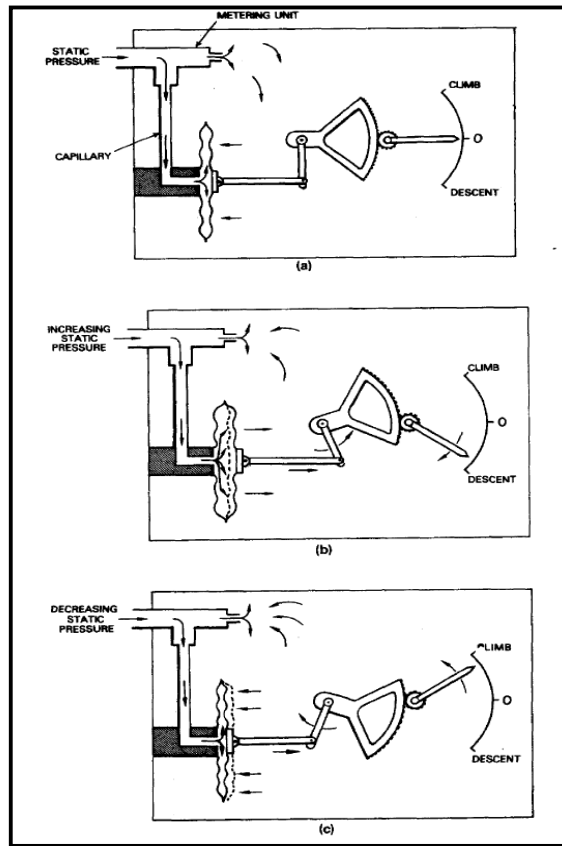
Figure (2.27): *Principle of vertical speed indicator.*

(a) Level flight: zero differential pressure across capsule.

(b) Descending: metering unit maintains case pressure lower than capsule pressure, changing it at the same rate and thereby creating a constant differential pressure across the capsule.

(c) Climbing: metering unit creates a constant differential pressure across capsule by maintaining case pressure higher than capsule pressure

2.3.1.6. Central Air Data Computers

The pressures on which the operation of the primary flight instruments is dependent are transmitted through a system of pipelines. It should be apparent therefore, that the length and quantity of the pipelines will vary according to the size of the aircraft, and also on the number of stations at which indications of the relevant air data are required. In order to minimize the 'plumbing' arrangements, the concept of supplying the pressures to a special unit at some centralized location, and then transmitting the air data

electrically to wherever required, was developed and resulted in the design of units designated as central aid data computers (CADC). Basically it's a device that produces electrical signal equivalents of pitot and static pressures by the combined operation of mechanical and synchronous transmission devices. The final computed output signals are then supplied to the appropriate indicators. The modular arrangement based on a representative (CADC) is shown schematically in Figure (2.28).
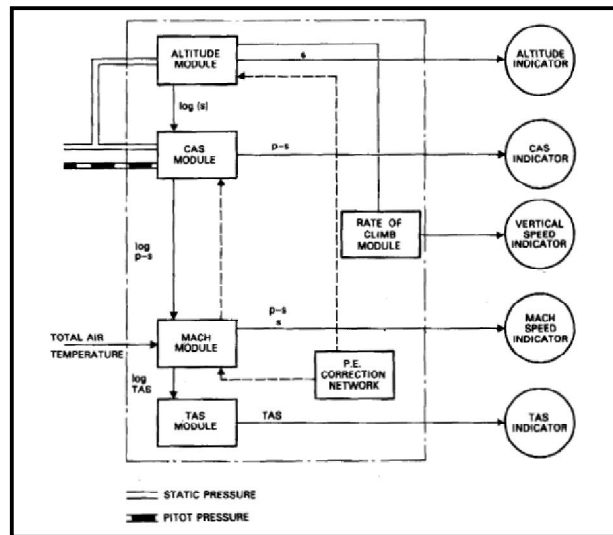
Figure (2.28): *Modular arrangement of a CADC system*

Pressure sensing is accomplished by two pressure transducers, one sensing static pressure within the altitude module, while the other senses both pitot and static pressures within the computed airspeed (CAS) module. The Mach speed module and true airspeed (TAS) module are pure signal-generating devices, which are supplied with altitude and airspeed signal data from the respective modules. Static air temperature data required for (TAS) computation is sensed by a probe located outside the aircraft, and is routed through the Mach speed module.

True airspeed (TAS) basically $= a \times M$ and computed according to the formula: $\text{TAS} = C.f(M).\sqrt{T_i}$ where:

- $f(M)$ is equal to $\sqrt{M^2/(1+0.2KM^2)}$
- Ti = absolute indicated stagnation temperature.

- K = a constant (the recovery factor of the temperature-sensing element).

2.3.2. Attitude instruments

The attitude instruments provide the necessary indications of the pitch, bank and turn attitudes of the aircraft. Attitude instruments are dependent on gyroscopic properties. A gyroscope is a mechanical system containing a heavy metal wheel, or rotor, universally mounted so that it has three degrees of freedom:

(i) *Spinning freedom* about an axis perpendicular through its centre (axis of spin XX).

(ii) *Tilting freedom* about a horizontal axis at right angles to the spin axis (axis of tilt YY).

(iii) *Veering freedom* about a vertical axis perpendicular to both the spin and tilt axes (axis of veer ZZ).

The three degrees of freedom are obtained by mounting the rotor in two concentrically pivoted rings, called inner and outer gimbal rings. The whole assembly is known as the *gimbal system of a free* or *space gyroscope*. The gimbal system is mounted in a frame as shown in Figure (2.28). The gyroscope possesses two important fundamental properties *gyroscopic inertia* or *rigidity,* and *precession***.** The two properties defined as follows:

- *Rigidity*, the property which resists any force tending to change the plane of rotation of its rotor. This property is dependent on three factors, the mass of the rotor, the speed of rotation, and the distance at which the mass acts from the centre.

- *Precession*, the angular change in direction of the plane of rotation under the influence of an applied force. The change in direction takes place, not in line with the applied force, but always at a point 90° away in the direction of rotation. The rate of precession also

depends on three factors, the strength and direction of the applied force, the moment of inertia of the rotor, and the angular velocity of the rotor.
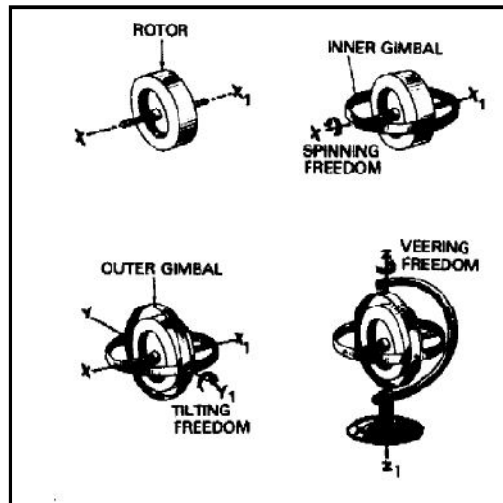
Figure (2.28): *Elements of a gyroscope*

2.3.2.1. References Established By Gyroscopes

For use in aircraft, gyroscopes must establish two essential references datums: A reference against which pitch and roll attitude changes may be detected, and a directional reference against which changes about the vertical axis may be detected. These references are established by gyroscopes having their spin axes arranged vertically and horizontally respectively, as shown in Figure (2.29).

The pitch, roll, and directional attitudes of the aircraft are determined by its displacement with respect to each appropriate gyroscope. For this reason, therefore, the gyroscopes are referred to as displacement type gyroscopes.

There are two principal methods used for driving the rotors of gyroscopic flight instruments: pneumatic and electric. In the pneumatic method the case of an instrument is connected to either an engine-driven vacuum pump, or a venturi located externally and in the slipstream of a propeller.

The high-performance gyroscopic instruments were designed for operation from aircraft electrical systems using greater rotor speeds, thus providing greater rigidity and stability of indications.
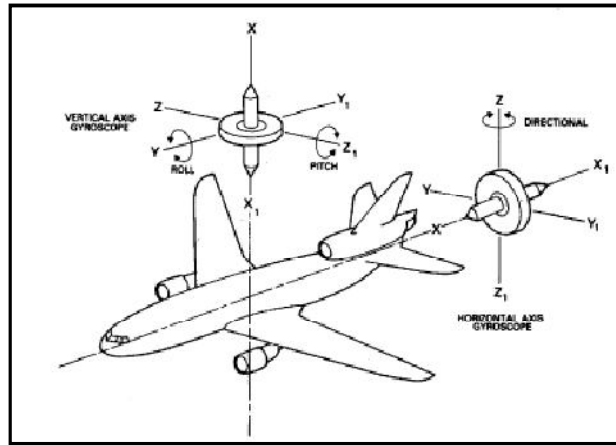
Figure (2.29): *Reference established by gyroscope*

2.3.2.2. The Gyro Horizon

The *gyro* horizon, or *artificial horizon*, indicates the pitch and bank attitude of an aircraft relative to the vertical, and for this purpose employs a displacement gyroscope, whose spin axis is maintained vertical by a gravity-sensing device.

Indications of pitch and bank attitude are presented by the relative positions of two elements, one symbolizing the aircraft itself, and the other in the form of a bar stabilized by the gyroscope and symbolizing the natural horizon. Supplementary indications of bank are presented by the position of a pointer, also gyro-stabilized, and a fixed bank angle scale. Two methods of presentation are shown in Figure (2.30) and the construction of the instrument is shown in Figure (2.31).

The gimbal system is arranged so that the inner ring forms the rotor casing, and is pivoted parallel to the aircraft's lateral axis YY_1 ; and the outer ring is pivoted parallel to the aircraft's longitudinal axis ZZ_1. The outer ring pivots are located at the front and rear ends of the instrument case. The element symbolizing the aircraft may be either rigidly fixed to the case, or externally adjusted up and down for pitch trim setting.
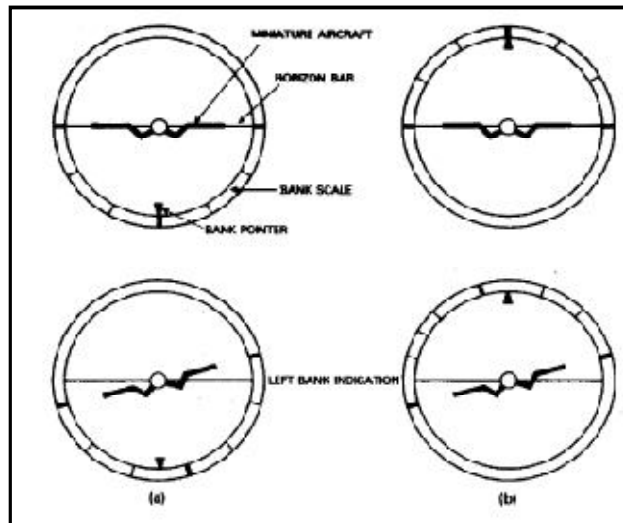
Figure (2.30): *Gyro horizon presentation*

(a) Bottom bank scale (b) Top bank scale
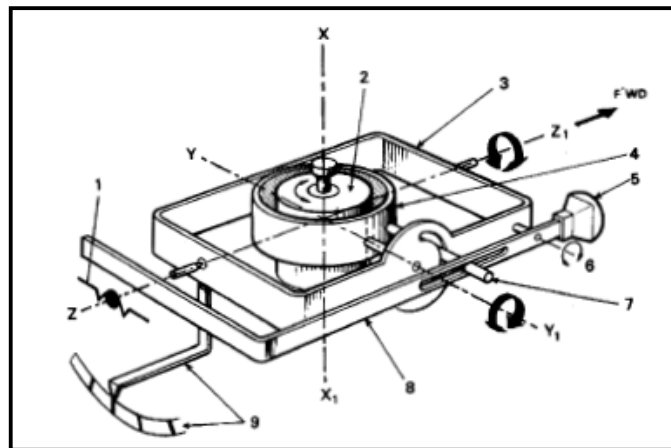
Figure (2.31): *A typical simple-type gyro horizon*

In operation the gimbal system is stabilized so that in level flight the three axes are mutually at right angles. When the aircraft goes into a climb, the instrument case and outer ring will turn about the axis YY_1, of the stabilized inner ring. The horizon bar is pivoted at the side and to the rear of the outer ring, and engages an actuating pin fixed to the inner ring, thus forming a magnifying lever system. In a climb attitude the bar pivot carries the rear end of the bar upwards causing it to pivot about the stabilized actuating pin. The front end of the bar and the pointer therefore move downwards through a greater angle than that of the outer ring, and since

movement is relative to the symbolic aircraft element, a climbing attitude is indicated.

Changes in the lateral attitude of the aircraft for banking, displaces the instrument case about the axis ZZ_1 and the whole stabilized gimbal system. Hence, lateral attitude changes are indicated by movement of the symbolic aircraft element relative to the horizon bar, and also by relative movement between the bank angle scale and the pointer.

Many modern aircraft currently in service employ integrated flight systems or *flight director systems* which comprise indicators that can display pitch and roll attitude data from remotely located vertical gyroscope units with associated signal data from radio navigation systems. Thus, the gyro horizon role is standby attitude indicator.

2.3.2.3. Rate Gyroscope

For the detection of rates of turn, direct use is made of gyroscopic precession; the gyroscope is arranged in the manner shown in Figure (2.32). Such an arrangement is known as a rate gyroscope. It has only one gimbal ring and it has a spring connected between the gimbal ring and casing to restrain movement about the longitudinal axis YY_1.

When the instrument is in its normal operating position, due to the spring restraint the rotor spin axis will always be horizontal and the turn pointer will be at the zero datum mark. With the rotor spinning, its rigidity will further ensure that the zero condition is maintained.
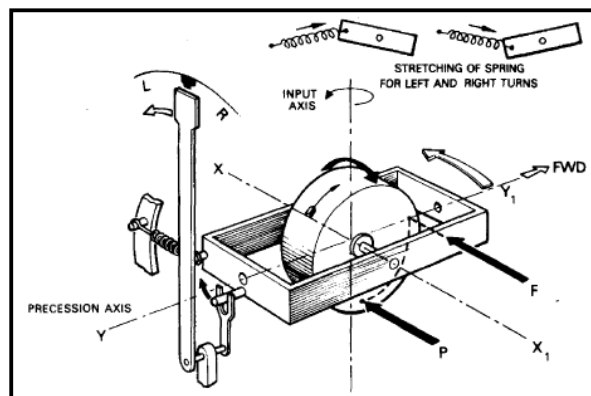
Figure (2.32): *Typical rate gyroscope*

In left turn about the vertical axis, the gimbal ring will also turn, but as the rigidity of the gyroscope resists this turning movement it will precess about axis YY₁. The direction of precession may be determined by the simple rule. A turn to the left causes a force to be applied at the front pivot of the gimbal ring, and this is the same as trying to push the rotor round at the point (F) on its rim. In following this through 90° in the direction of rotation, precession will take place at point (P), thus causing the gimbal ring and rotor to tilt about the longitudinal axis. If a pointer were fixed to the gimbal ring, it would tilt through the same angle too, and would indicate a turn and also its direction. The angular deflection of the gimbal ring is controlled by connecting it to the instrument case through a spring; it will stretch the spring until the force it exerts prevents further deflection of the gyro. Since precession of this type of gyroscope is equal to the product of angular momentum of the gyroscope and the rate of turn, then the spring force is a measure of the rate of turn.

2.3.3. Bank Indication

In addition to the primary indication of turn rate, it is also necessary to have an indication that the aircraft is correctly banked for the particular turn. A secondary indicating mechanism is therefore provided which depends for its operation on the effect of gravitational and centrifugal forces. The principal mechanical method employed is a ball in a curved liquid-filled glass tube which illustrated schematically in Figure (2.33).

In normal flight, diagram (a), gravity holds the ball in center of the tube. At (b) the aircraft is shown turning to the left at a certain airspeed and bank angle. The indicator case move with the aircraft, and because of the turn, centrifugal force in addition to that of gravity acts upon the ball and tends to displace it outwards from the centre of the turn. However, when the turn is executed at the correct bank angle then there is a balanced condition between the two forces and so the ball remains at the center position.
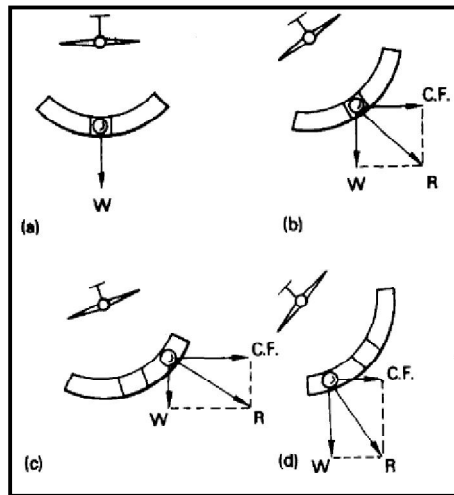
Figure (2.33): *Ball-type bank indicating element*

If the bank angle for a particular turn rate is not correct, say under banked as in diagram (c), then the aircraft will tend to skid out of the turn. Centrifugal force will predominate under such conditions and displace the ball from its center position. When the turn is over banked, as at (d), the aircraft will tend to slip into the turn and so the force due to gravity will now have the predominant effect on the ball and it will be displaced from center in the direction to that of an under banked turn.

2.3.4.   Heading indicating Instruments

The first of the many air-borne flight and navigational aids is the Direct-reading magnetic compasses, their primary function being to show the direction in which an aircraft is heading with respect to the earth's magnetic meridian. As far as present-day aircraft and navigational aids are concerned, however, such a directional reference is more accurately provided by remote-reading compass systems, and flight director systems and so direct-reading compasses are relegated to a standby role.

In basic form, remote-indicating compasses are systems in which a magnetic detecting element monitors a gyroscopic indicating element, the fundamental operating principles of the main components shown in the block diagram of Figure (2.34).
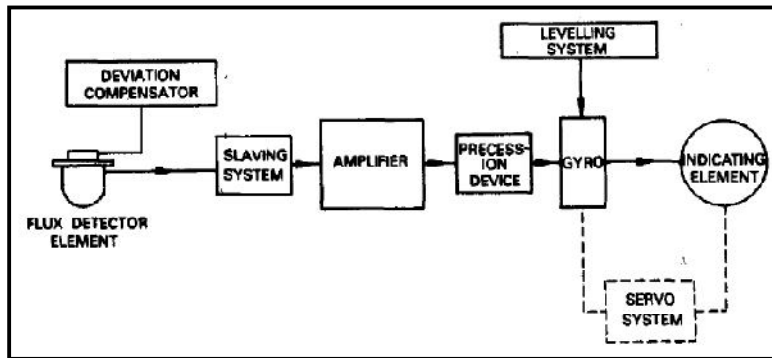
Figure (2.34): *Essential component of a remote-indicating compass system*

The detector element used in all remote-indicating compasses is of the fixed type which detects the effect of the earth's magnetic field as an electromagnetically induced voltage and monitors a heading indicator by means of a variable secondary output voltage signal. In other words, the detector acts as an alternating-current type of synchro transmitter.

Monitoring refers to the process of reproducing the directional references established by the flux detector element as quantitative indications on the heading indicator. The principle of monitoring illustrated in Figure (2.35).

When the flux detector is positioned steady on one heading (000°)*,* then a maximum voltage signal will be induced in the pick-off coil (A), while coils (B) and (C) will have signals of half the amplitude and opposing phase induced in them. These signals are fed to the corresponding tappings of the synchro receiver stator, and the fluxes produced combine to establish a resultant field across the centre of the stator. This resultant is in exact alignment with the resultant of the earth's field passing through the detector. If, as shown in Figure (2.35-a), the synchro receiver rotor is at right angles to the resultant flux then no voltage can be induced in the windings. In this position, the synchro is at 'null' and the directional gyro will also be aligned with the earth's resultant field vector and so the heading indicating element will indicate 000°.
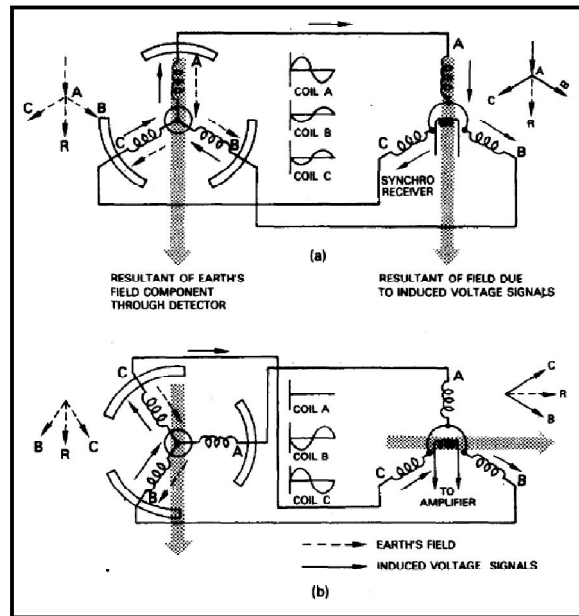
Figure (2.35): *Directional gyroscope monitoring*

(a) Heading = 000°   (b) Heading = 090°

When the flux detector element turns through 90°; the disposition of the detector pick-off coils will be as shown at (b). No signal voltage will be induced in coil (A), but coils (B) and (C) have increased voltage signals induced in them, the signal in (C) being opposite in phase to what it was the 'null' position. The resultant flux across the receiver synchro stator will have rotated through 90°, and assuming for a moment that the gyroscope and synchro are still at their original position, the flux will now be in line with the synchro rotor and will therefore induce maximum voltage in the rotor. This error voltage signal is fed to a slaving amplifier in which its phase is detected, and after amplification is fed to a slaving torque motor, which precesses the gyroscope and synchro rotor in the appropriate direction until the rotor reaches the 'null' position, once again at right angles to the resultant field. In other words, the conditions will be the same as shown in diagram (a) but on the new heading of 090°.

An example of the dial presentation of an indicator used with a master directional gyroscope element is shown in Figure (2.36). In addition to displaying magnetic heading data, it also displays the magnetic bearing of

an aircraft with respect to the ground stations of the radio navigation systems ADF (Automatic Direction Finding) and VOR (Very high-frequency Omnidirectional Range). For this reason, the indicator is generally referred to as a Radio Magnetic Indicator (RMI).

The bearing indications are provided by two concentrically mounted pointers, one called a 'double-bar' pointer and the other a 'single-bar' pointer. Both are referenced against the main compass card, and are positioned by synchros connected to the ADF and VOR receivers on board the aircraft.
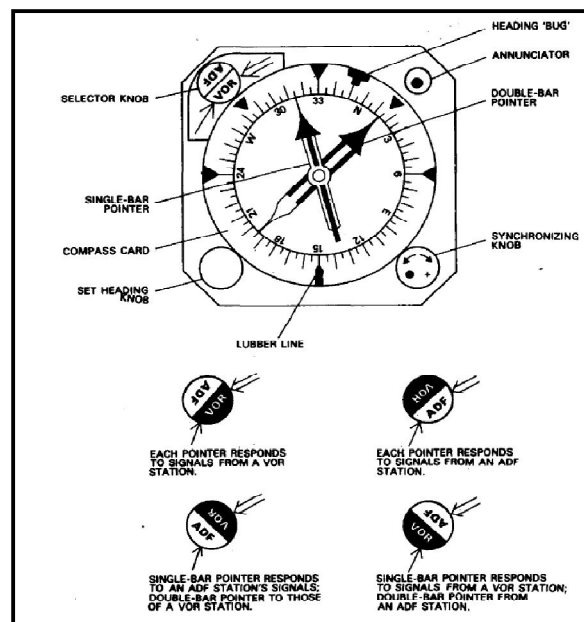
Figure (2.36): *Heading Indicator*

The display is controlled by manual adjustment of a selector knob to the positions shown in the diagram. The signals transmitted to the synchro systems are such that the pointers always point to the stations from which the signals are received. This may be seen from Figure (2.37) which is a representation of how the position of an aircraft may be determined from the three sources of navigational information.

In order that a pilot may set a heading on which it is desired for the aircraft to be flown, a 'set heading' knob is provided. It is mechanically coupled to a heading 'bug' so that when the knob is rotated, the 'bug' is also rotated

with respect to the compass card. The aircraft is then turned until the desired new heading is registered against the 'bug'.
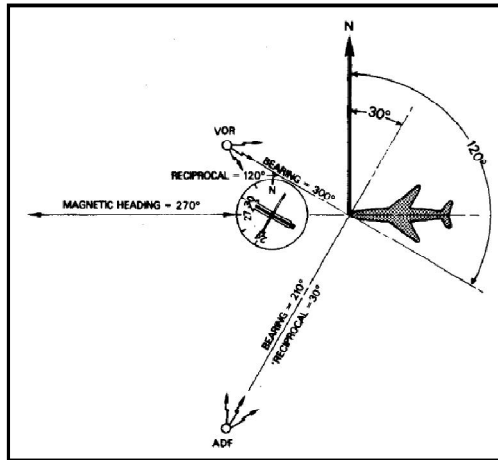
Figure (2.37): *RMI Indications*

2.3.5. Measurement of engine speed

The measurement of engine speed is of considerable importance, it permits an accurate control over the performance of the appropriate type of engine to be maintained.

With reciprocating engines the speed measured is that of the crankshaft, while with turboprop and turbojet engines the rotational speed of the compressor shaft is measured, such measurement serving as a useful indication of the thrust being produced. The indicating instruments are normally referred to as tachometers.

The generator and indicator electrical tachometer is the common system; the generator consists of a permanent-magnet rotor rotating within a slotted stator which carries a star-connected three-phase winding. The method most commonly used for driving a rotor is by means of a splined shaft coupling, the generator as a whole being bolted directly to a mounting pad at the appropriate accessories drive gear outlet of the engine.

As shown in Figure (2.38), when the generator rotor is driven round inside its stator, the poles sweep past each stator winding in succession so that three waves or phases of alternating e.m.f. are generated, the waves being

120° apart. The magnitude of the e.m.f. induced by the magnet depends on the strength of the magnet and the number of turns on the phase coils. Furthermore, as each coil is passed by a pair of rotor poles, the induced e.m.f. completes one cycle at a frequency determined by the rotational speed of the rotor, Therefore, rotor speed and frequency are directly proportional, and since the rotor is driven by the engine at some fixed ratio then the frequency of induced e.m.f. is a measure of the engine speed.
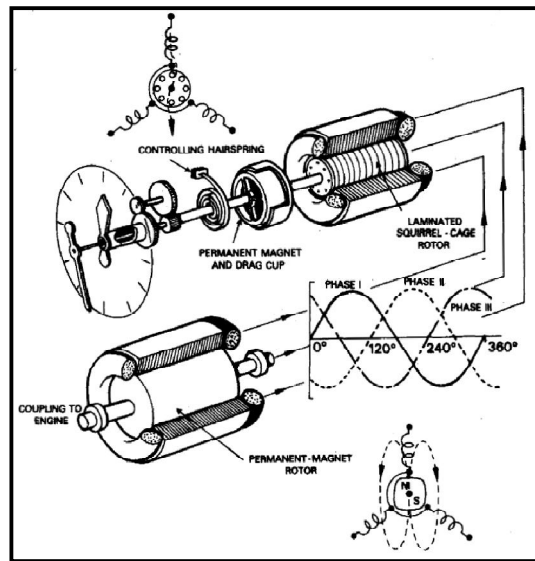
Figure (2.38): *Principle of tachometer's generator and indicator*

The e.m.f.'s generator are supplied to the corresponding phase coils of the indicator stator to produce currents of a magnitude and direction dependent on the e.m.f.'s. The distribution of stator currents produces a resultant magnetic field which rotates at a speed dependent on the generator frequency. As the field rotates it cuts through the copper bars of the squirrel-cage rotor, inducing a current in them which, in turn, sets up a magnetic field around each bar. The reaction of these fields with the main rotating field produces a torque on the rotor causing it to rotate in the same direction as the main field and at the same speed.

As the rotor rotates it drives the permanent magnet of the speed indicating unit, and because of relative motion between the magnet and the drag-cup eddy currents are induced in the latter. These currents create a magnetic

field which reacts with the permanent magnetic field, and since there is always a tendency to oppose the creation of induced currents (Lenz's law), the torque reaction of the fields causes the drag-cup to be continuously rotated in the same direction as the magnet. However, this rotation of the drag-cup is restricted by the calibrated hairspring in such a manner that the cup will move to a position at which the eddy-current-drag torque is balanced by the tension of the spring. The resulting movement of the drag-cup shaft and gear train thus positions the pointers over the dial to indicate the engine speed prevailing at that instant.

The measurement of engine speed in terms of a percentage is adopted for turbojet engine operation, and was introduced so that various types of engine could be operated on the same basis of comparison. The dial scales are calibrated from 0 to 100%, with 100% corresponding to the optimum turbine speed.

In aircraft powered by a multi-arrangement of engines, the problem arises of maintaining the engine speeds in synchronism at 'on-speed' conditions and so minimizing the effects of structural vibration and noise. In order to facilitate manual adjustment of multi-engines speeds an additional instrument known as a *synchroscope* was introduced. It provides a qualitative indication of the differences in speeds between two or more engines, and by using the technique of setting up the required on-speed conditions on a selected master engine, the instrument also provides a clear and unmistakable indication of whether a slave engine is running faster or slower than the master.

The instrument was designed at the outset for operation from the alternating current generated by the tachometer system, and it therefore forms an electrical part of this system. The dial presentations of a combination dual r.p.m. and synchroscope presentation is shown in Figure (2.39).
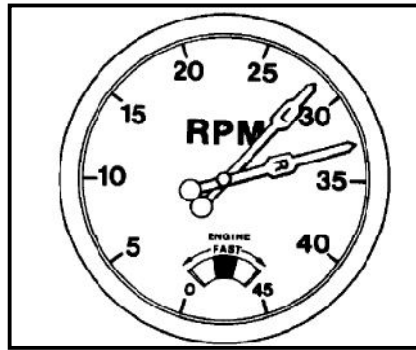
Figure (2.39): *Combined dual tachometer and synchroscope*

2.4. Aircraft Integrated Data Systems (AIDS)

The recording of certain of the parameters measured by instruments, and of any unusual incidents connected with engines, airframes and performance generally, has always been a feature of in-flight management procedures. Therefore, it became mandatory to equip certain categories of aircraft with an automatic flight recording system comprising a flight data recorder (FDR) and a cockpit voice recorder (CVR). The number of parameters to be recorded progressively increases from a common basic set established according to groups of aircraft weights, and therefore according to the degree of complexity of aircraft and systems. Since the majority of the parameters are measured by 'tapping off' from the sensing elements or transducers of existing instrument systems then, consistent with the number of these employed in an aircraft, there is virtually no limit to the number of parameters that can be recorded. Thus, the capacity of recorders can be increased to cater not only for the data required by the regulations (mandatory recording) but also for the acquisition of data which can be of value to an aircraft operator in planning for economic operation, and for the monitoring of aircraft's engines and systems reliability (non-mandatory recording) and even for specialists training. Recording systems of this nature utilize two complementary recorders and are known variously as Aircraft Integrated Data Systems (AIDS) or Data Acquisition Systems (DAS).

2.4.1. Flight Data Recorders (FDR)

The mandatory requirements for aircraft to be equipped with flight data recorders relate primarily to the acquisition of data which will prove valuable during investigations into the cause of a crash. Thus, the overall development of data recording systems has been built up using such data acquisition as the foundation.

The relative value of flight recording has, from investigators' experience, been indicated in respect of two very general categories of crash. The first category is that in which some form of aircraft, or system, malfunction has been the primary cause. The second category is that in which the crash has an operational cause. Some examples of this are loss of control resulting from improper handling by the flight crew, adverse weather conditions, or navigational factors.

Table (2.1): *Evolution of flight data collection devices*

FDR Type	Introduced into service	Number of parameters	FDR data capacity
Analogue	1958	5	Mechanical limit of about 10 Parameters
Digital (tape medium)	1993	280	128 wps (serial data input)
Digital (solid-state) combi-recorder	2004	774	256 wps (serial data input)
Digital (solid-state)	2007	> 1,000	1,024 wps (serial data input)
Digital (solid-state) EAFR (stores both cockpit audio and flight data)	2009	> 1,000	Ethernet system

2.4.1.1. Parameters selected

In the selection of the mandatory parameters for crash investigation purposes, the objective is to obtain either directly or by deduction from the recorded data, the following information:

1. The aircraft's flight path and attitude in achieving that path.
2. The basic forces acting on the aircraft, e.g. lift, drag, thrust and control forces.

3. The general origin of the basic forces and influences, e.g. navigational information, aircraft system status information.

The mandatory parameters are generally related to the following:

1. Time (GMT or elapsed).

2. Pressure altitude.

3. Airspeed.

4. Vertical (normal) acceleration.

5. Magnetic heading.

6. Pitch attitude.

2.4.1.2. Electro-magnetic recording

The FDR with electro-magnetic recording method uses the analogue signal data from the transducers appropriate to each parameter are first encoded, transformed into a digital coded form in accordance with the binary scale of notation. The encoded signals then pass through such sub-channels as conditioning, logic, multiplexing, and modulation, and are fed as a series of pulse currents to an electromagnetic recording or writing head, the currents being of either positive or negative polarity and corresponding respectively to a '1 state' or a '0 state'. The magnetic core of the writing head has a very small air gap (typically 0.001 inch) between pole pieces, and is located in very close proximity to the recording medium which, in some recorders, may be a plastic-based tape coated with ferrite material, while in others, it is stainless steel wire.

The tape or wire is wound on spools and guide rollers, and is transported over the writing head by a synchronous motor supplied with a 115 Va.c. The transport rate is typically 1.75 inch per second for magnetic tape and 2.5 inch per second for wire. An example of one method of transporting magnetic tape is illustrated in Figure (2.40).

As shown in Figure (2.41), a pulse current assigned the value equivalent to binary state '1' will magnetize with a positive polarity, while a pulse

current equivalent to binary state '0' will magnetize with a negative polarity.
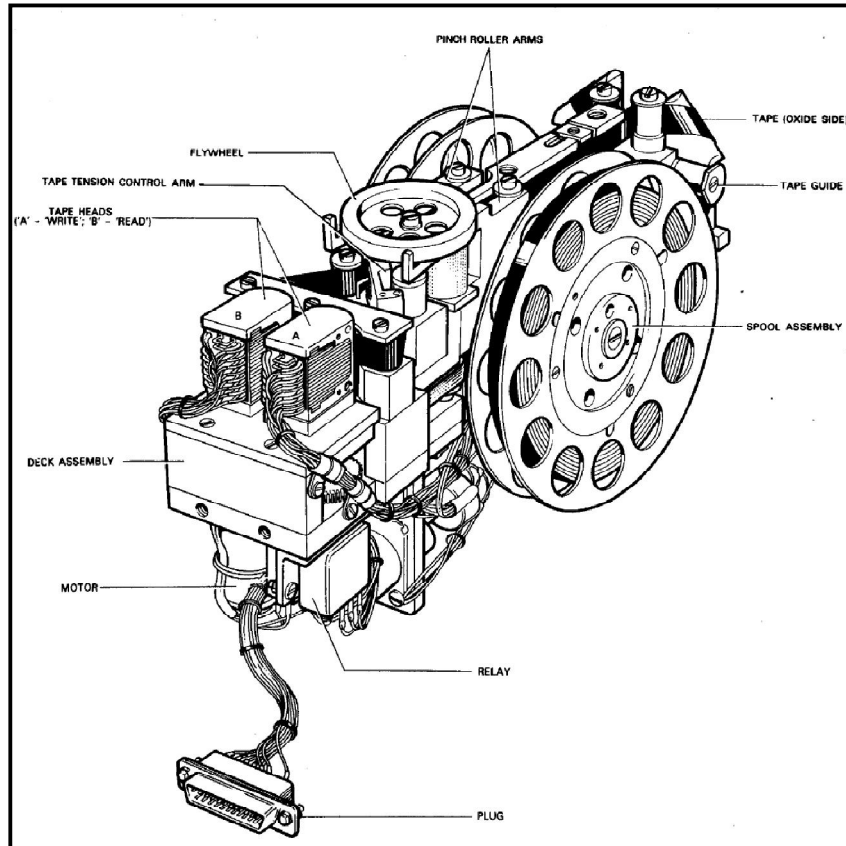
Figure (2.40): *Example of transporting magnetic tape*
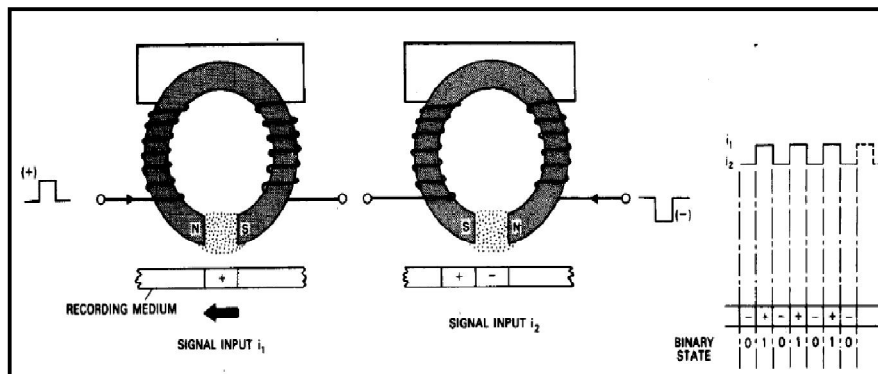
Figure (2.41): *Writing head operation*

A number of magnetize bits corresponding to the parameters measured comprise a word. Since the words are polarized on the tape or wire, then the latter provide what is termed a magnetic store of information. The number of bits which can be stored in a unit length of tape or wire gives

the packing density; some typical values are 900 bits per inch for tape, and 414 bits per inch for wire. The technique of electro-magnetic recording is illustrated in Figure (2.42).
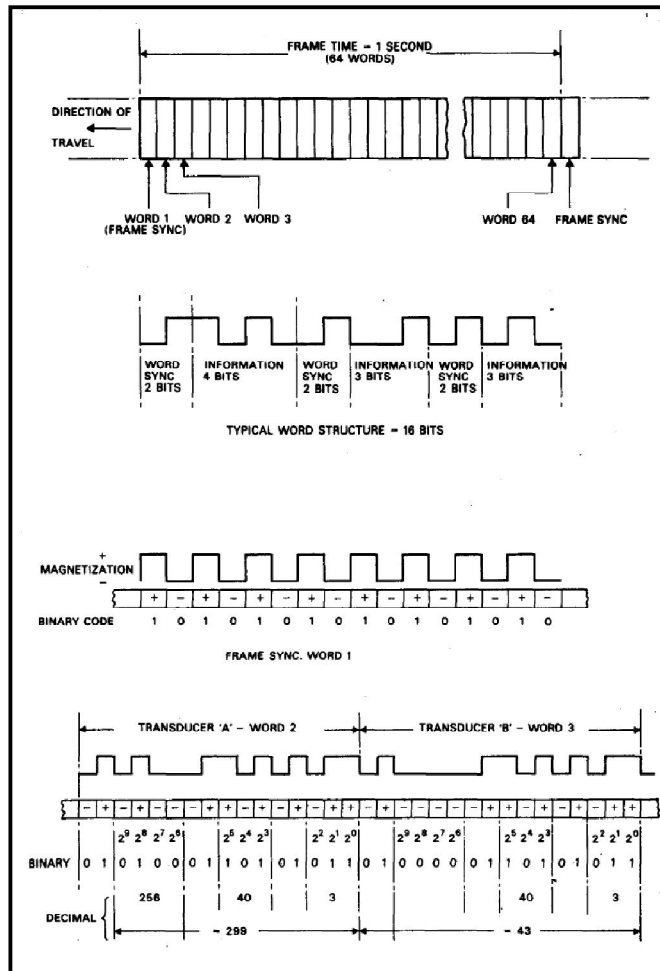
Figure (2.42): *Electromagnetic recording*

To provide correct interpretation of the stored information when reading the tape or wire, the signal inputs to the magnetizing coils of the writing head must first be assembled into an identifiable format. This is done by multiplexing the pulses in a signal conditioning unit, so that the format comprises a frame of 64 words, each word containing 16 bits; a frame is of (l) second duration. The first word in each frame is a frame synchronizing signal comprising a repetitive series of '1s' and '0s', the remaining 63 words being available for pure information on the required parameters.

Each of these words include three interposed two-bit (0,l) word synchronizing signals which divide the pure information into groups of 4 bits, 3 bits and 3 bits respectively. Thus, each word has an equivalent full-scale decimal count of 1024. In the example shown in Figure (2.42), the ten information bits of word 2 from a transducer 'A' have been magnetically encoded (0100,101,011) so that the code therefore converts to 256 + 40 + 3 and equal to 299. Similarly, the word 3 from transducer 'B' converts to 43 in decimal notation. Information on all parameters is sampled within the signal conditioning unit of the flight recorder system at periodic intervals to construct the multiplex pattern, some parameters being sampled more frequently than others. Some details of sampling rate and word allocation based on one type of wire recorder in current use are given in Table (2.2).

Table (2.2): *Sampling rate and word allocation*

Parameter	Sampling rate (per second)	Word number allocation
Time	1	1
Acceleration	8	6, 14, 22, 30, 38, 46, 54, 62
Pitch	8	2, 10, 18, 26, 34, 42, 50, 58
Altitude	1	9
Airspeed	1	13
Heading	1	29
Flight number and date in three groups of 10; 1 at ➔ and 2 at ➔	2 } 1 }	3, 5, 35, 37

For identification purposes, it is necessary to record flight reference information, i.e. flight number and date. This is done by operating 'thumbwheel' switches on a flight encoder panel located in the flight crew compartment. The switches are numbered and when selected they feed an

output to the signal conditioning unit for multiplexing and transmission to the recorder in groups of binary coded information.

2.4.2. Basics of aircraft integrated data systems (AIDS)

The capacity of recorders can be increased to acquire data additional to that associated with accident investigation. Except for the requirements laid down for certain certification processes, the acquisition of such data as airframe and engine systems operation, is non-mandatory. It does, however, provide an operator with the means of monitoring the performance of systems on a comparative basis, and from any variations to obtain advance warnings of possible failures of a component or a system overall.

Systems of this nature which are known variously by the acronyms AIDS (Aircraft Integrated Data System) and DAS (Data Acquisition System) may be used separately, or in conjunction with an accident recorder. The schematic arrangement of a system is shown in Figure (2.43).
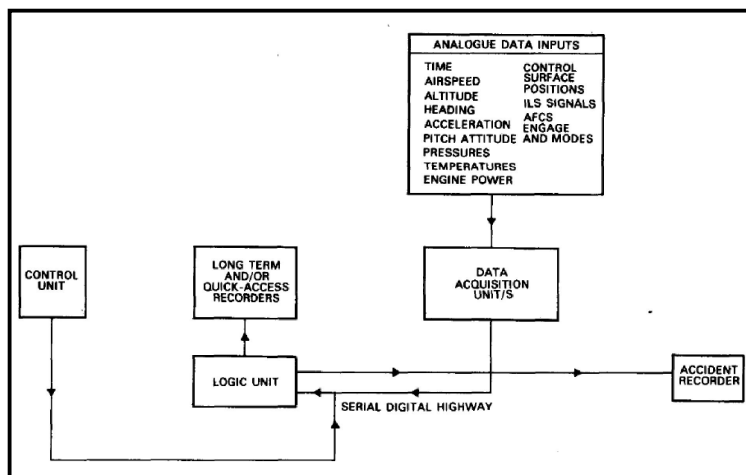
Figure (4.5): *Schematic arrangement of an integrated data system.*

The analogue inputs from the appropriate transducers are supplied to one or more data acquisition units each containing analogue and digital multiplexers, signal conditioning circuits, and an analogue-to-digital converter. After processing and conversion, the signals are 'serialized' and transmitted to the logic unit along what is termed a digital highway; this

provides a common data link for all the acquisition units which may make up a system, and so keep interconnection cabling to a minimum. The logic unit converts the information from the acquisition units into the particular formats appropriate to the various output requirements. The control unit enables the flight crew to insert flight information into the record, and also provides for continuous system monitoring and simple test facilities.

The recorders used in these systems are normally called Solid State Flight Data Recorder (SSFDR). Figure (2.44) shows internal design of a typical SSFDR. "Solid State" refers to storage of data in semiconductor memories or integrated circuits, rather than using the older technology of electromechanical methods of data retention. SSFDRs can record more than 700 parameters or seven times the capability of magnetic-tape recorders.
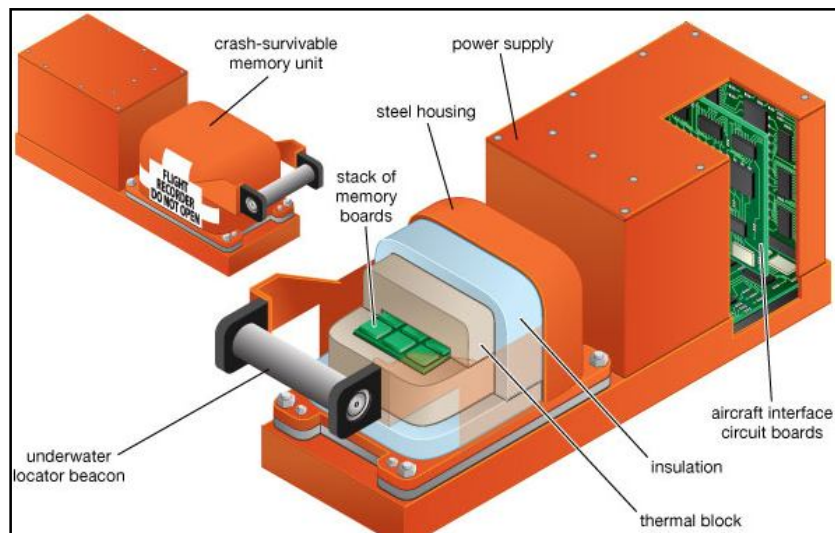
Figure (2.44): *Internal design of a typical SSFDR*

2.4.3. Protection and location of flight recorders

Flight recorders for the retrieval of data for accident investigation purposes must be adequately protected and so located in an aircraft that the recording media will survive in the event of a crash situation. The adverse factors against which protection is necessary are mechanical damage due to impact and acceleration forces, fire damage, and chemical attack by

hydraulic, de-icing and fire extinguishing fluids, fuels and acids. Protection must also be afforded against the effects of sea water, and to cater for cases where an accident occurs resulting in wreckage entering deep water, the recording medium container must also be provided with separation and flotation facilities.

The storage medium of each recorder is located in a protective capsule, which must be able to withstand an impact of 3,400 Gs (3,400 times the force of gravity). Additionally, each must also survive flames at 2,000 degrees Fahrenheit for up to 30 minutes, and submersion in 20,000 feet of saltwater for 30 days. Most flight data recorders are constructed with an aluminum sheet metal chassis, with a protective capsule constructed of heat-treated stainless steel or titanium. In many accidents, the only devices that survive are the Crash Survivable Memory Units (CSMU) of the flight data recorders and cockpit voice recorders. Typically, the rest of the recorders' chassis and inner components are mangled. The CSMU is a large cylinder that bolts onto the flat portion of the recorder. This device is engineered to withstand extreme heat, violent crashes and tons of pressure. In older magnetic-tape recorders, the CSMU is inside a rectangular box.

The location of a recorder in an aircraft is an important factor, while the rear fuselage and tail unit structure is the section most likely to survive, therefore, a protected flight data recorder is installed in the rear of an aircraft.

In addition to the foregoing protection and location requirements, it is also necessary to provide recorders with means to facilitate their identification in conditions following a crash. This is usually done by adopting a distinctive colour scheme; thus, contrary to the 'black box' cliche frequently adopted in news media reports of a crash, the box could well be coloured fluorescent orange. In addition, FDRs are equipped with an Underwater Locator Beacon (ULB) to help locate the box after a water

crash. If a plane crashes into the water, this beacon sends out an ultrasonic pulse that detectable by sonar and acoustical locating equipment. There is a submergence sensor on the side of the beacon, when water touches this sensor, it activates the beacon. The beacon sends out pulses at 37.5 kilohertz (kHz) and can transmit sound as deep as 14,000 feet (4,267 m). Once the beacon begins "pinging," it pings once per second for 30 days. This beacon is powered by a battery that has a shelf life of six years.

2.4.4. Data recovery and analysis

The handling of data derived from a flight recorder system falls into two distinct phases: the recovery of raw data from the recording medium, and analysis of the information in relation to a particular accident or, in the case of data acquisition systems, in relation to data sampling for the purpose of routine monitoring. In addition to the recovery of raw data, the first phase also includes the application of corrections for calibration and other system errors, e.g. pressure errors of a pitot-static system, and their presentation in a form suitable for computation and analysis. The second phase requires the introduction of other information such as design, performance or operational data, relevant to the type of aircraft.

The nature of the equipment required for data recovery depends on the recording method adopted.

2.5. Тестер У3-ЛК flight data recorder

Тестер У3-ЛК airborne FDR is a Russian flight data recorder used widely in the Soviet Union aircrafts which is about 80% of the operated airplane in Sudan. Тестер is intended for recording code-pulse information on a magnetic tape during the flight and keep the recorded data in normal and emergency flight condition for the subsequences processing on the ground.

2.5.1. System Specifications:

The characteristics of the system illustrated in Table (2.3). The data accumulated by this recorder can be used for the following purposes:-

a) Monitoring of pilot actions.

b) Determination of flight performance limitation exceeding.

c) Monitoring of aircraft operation status and technical condition to improve safety.

d) Investigation of air accidents and their potential factors.
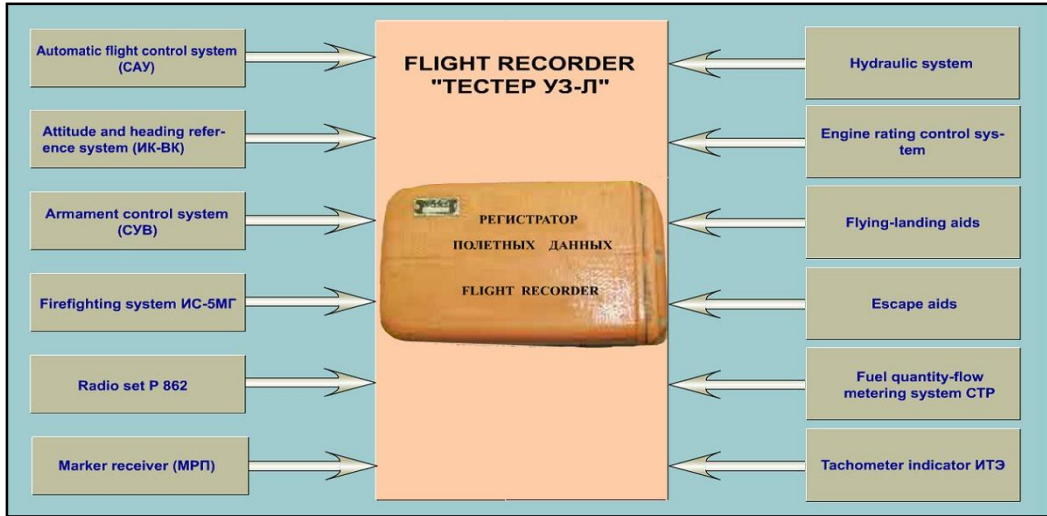
Figure (2.45): *Interconnection of Тестер У3-ЛК airborne FDR*

Table (2.3): *characteristics of the Тестер У3-ЛК system*

Characteristic Name	Value
Data Safety	Last three hours (Earlier recorder data will be erased automatically)
Time of recorder continuous operation (RECORD Mode)	24 Hours
Time of readiness for operation at temperature higher than -20°C	Not more than 1.5 minutes
Power consumption (V = 27V)	Not more than 70 W
Limit of power supply changing	18 -31V
Recorder mass	Not exceed 21Kg
Storage of recorded data without losses or changes	During 60 days

Data security at exposure of gasoline, kerosene, all hydraulic and fire-extinguisher liquids	During 2 days
Data security at exposure of Sea water	During 5 days
Data security at exposure of temperature 1,000°C	During 15 minutes
Data security at exposure of G-load equal to 200	During 10ms

2.5.2. System components

The Тестер У3-ЛК system is a combination of units interconnected through electrical connectors as shown in Figure (2.46), these units are:
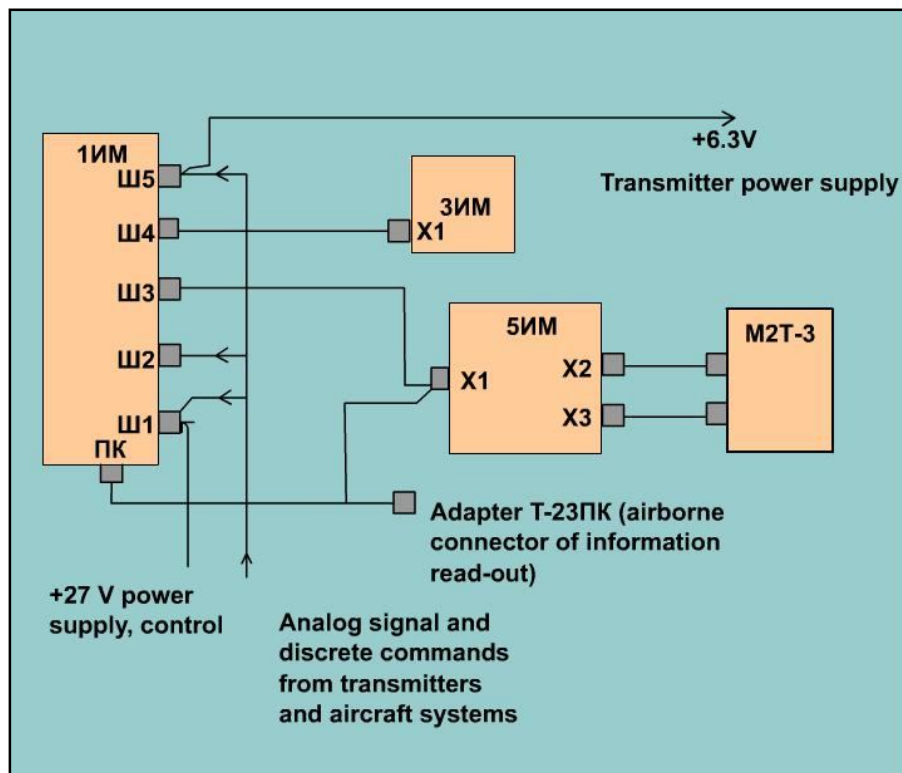
Figure (2.46): *Electrical diagram of Testep system units*

2.5.2.1. Electrical unit (1 ИМ),

This unit intended for:

a. Reception of Ux signals $(0 - 6.3V)$ from airborne systems sensors, F signals $(7 - 100$ Hz) and discrete command signals (pulses with duration $(t > 3ms)$ and $(18 - 31V)$).

b. Convertion of electrical signals into 8-bit (for Ux), 14-bit (for F) and 7-bit (for discrete commands).

c. Send codes corresponding to the electrical values to the recorder.

d. Reception information from the Switching number unit and control the system operation according to this information.

Figure (2.47): *Electrical unit (1 ИМ)*

2.5.2.2. Switching Number unit (3 ИМ)

This unit intended for automatic determination of flight parameters for recording on the magnetic tape.
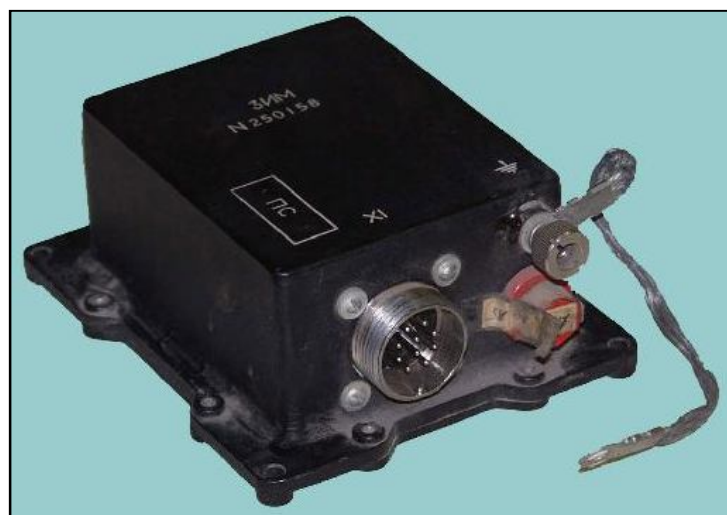
Figure (2.48): *Switching Number unit (3 ИМ)*

2.5.2.3. Recording, Playback and Monitoring unit (5 ИМ)

This unit intended for:

a. Amplification and communication of data pulses for/from recording/playback.

b. Control of magnetic media movement.

c. Drive the electrical motor in two modes (Recording/Playback).

Figure (2.49): *Recording, Playback and Monitoring unit (5 ИМ)*

2.5.2.4. Magnetic Tape Recorder unit (M2T-3-3)

This unit intended for:

a. Recording of code-pulse data on magnetic tape.

b. Playback of the recorded data and self-test operations.

c. Storage of magnetic tape with the recorded data in case of accident.

Figure (2.50): *Magnetic Tape Recorder unit (M2T-3-3)*
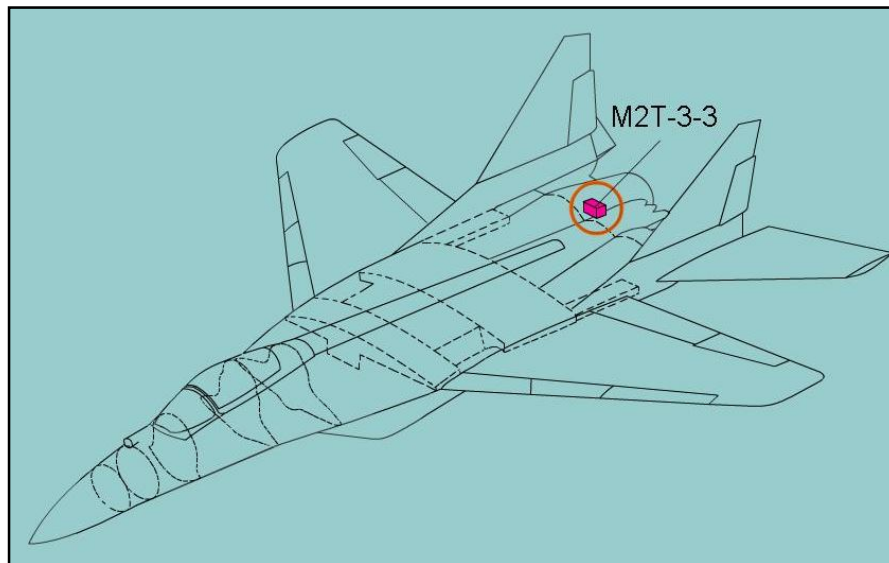
Figure (2.51): *Position of the Magnetic Tape Recorder unit (M2T-3-3)*

2.5.3. System operation

The Тестер У3-ЛК system operates manually or automatically. The manual operation performed on the ground by switching on the (RECORDER) switch located on the right-side panel in the cockpit. The automatic operation is effectuated during extension of the flaps when the RPM of the right engine is ($N_k > 85\%$) and as the aircraft lifts off. Figure (4.14) shows the manual and automatic operations.

2.5.4. Transfer information to the ground decoder

Transfer information from the magnetic tape recorder unit (M2T-3-3) to the ground decoder device is done in compliance with the task card of the aircraft Maintenance Manual with the use of (T-23LK) adapter to save information into a flash-card. This operation includes:

    a. Prepare the ground recorder for information transfer.

    b. Switch on the (REGISTER) switch on the right-side panel in the cockpit.

    c. Connect the recorder unit to the on board connector.

    d. Record information in the Direct Read-out mode task card.
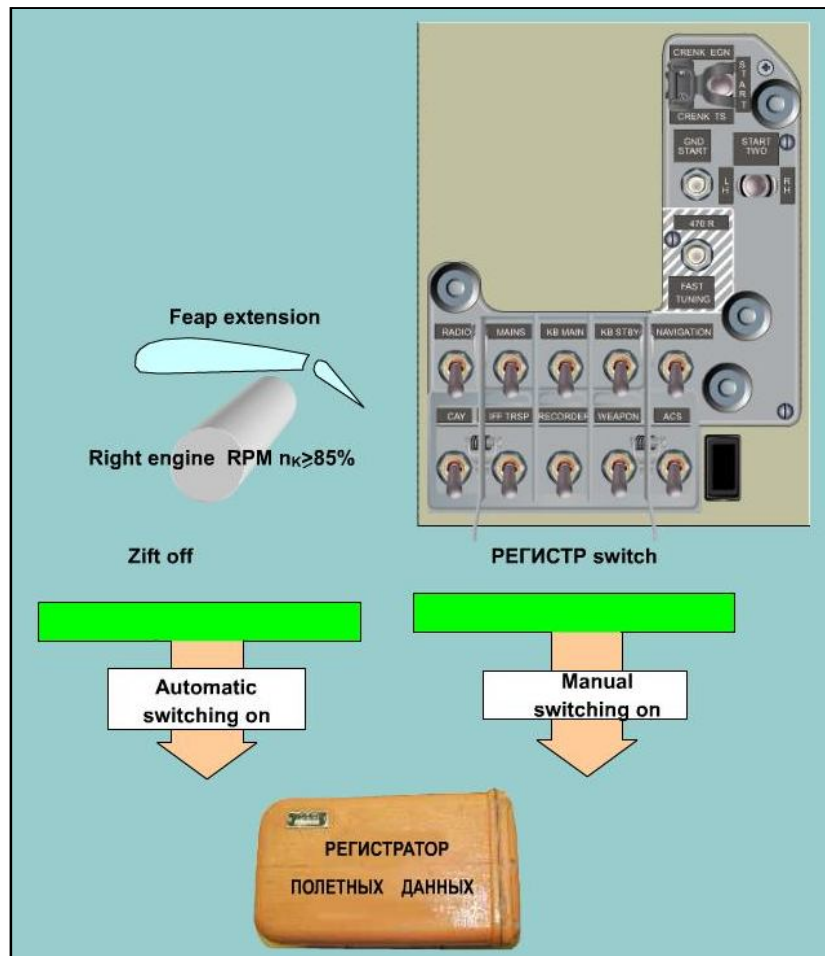
Figure (2.52): *The Tectep manual and automatic operations*

Chapter Three
Flight Simulation

3.1. Introduction

Simulation of a flight session and reproduction of the flight path from recorded flight parameters in flight data recorder (FDR), is gone through many phases of complex, basing on aerodynamics and theory of flight, keeping track of aircraft motion about one or more of three axes at the response to the flight controls, and its relationship to the aircraft's speed in its different forms and altitudes. The aircraft movement is transformed in form of parameters measured by the instruments, which are designed in different ways to meet the nature of the parameter to be measured, such as the speed of response and the limits of measurement and other properties. Then, the simulation process is gone through analysis of hundreds of recorded flight parameters according to time factor in the flight recorder. Therefore, to classify, identify and extract the parameters that affect the movement of the aircraft and the flight track.

The programming and writing code for simulation is established with the Microsoft Flight Simulator properties and methods to access the aircraft parameters that are used by the flight simulator comparing them to those extracted from the flight data recorder. Then, reforming the parameters of the flight data recorder to meet the specifications of the Microsoft Flight Simulator parameters in order to pass the flight parameters through the FSUIPC (Flight Simulator Universal Inter-process Communication) to the flight simulator to simulate the aircraft behavior and motion during the real flight session.

3.2. FDR data recovery and analysis

As stated before, handling of data derived from a flight recorder system falls into two distinct phases: the recovery of raw data from the recording medium, and analysis of the information in relation to a particular

purpose. Transfer informations from the magnetic tape recorder unit (M2T-3-3) of Тестер У3-ЛК flight data recorder to the ground decoder device is done in compliance with the task card of the aircraft Maintenance Manual with the use of (T-23LK) adapter to save information into a flash-card. This operation includes:

1. Prepare the ground recorder for information transfer.
2. Switch on the (REGISTER) switch on the right-side panel in the cockpit.
3. Connect the recorder unit to the on board connector.
4. Record information in the Direct Read-out mode task card.

Then, the flight data are transferred to Dvina-M software to decode the data and to extract the parameters and their values which are recorded each one second. Dvina-M software produces the parameters data in two forms: tables saved as three text files for each flight session, and charts are saved as (.trf) files.
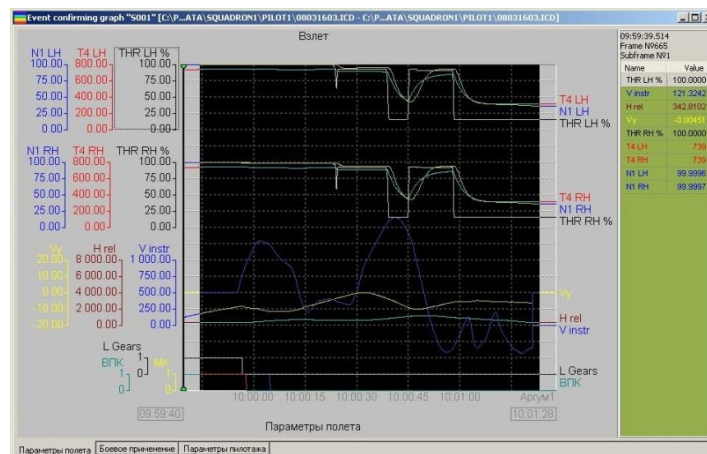
Figure (3.1)*: Dvina-M software charts.*

Figure (3.2): *Dvina-M software tables.*

The total number of parameters is saved by the Тестер УЗ-ЛК flight data recorder is 125 parameters. 90% of the parameters are relevant to aircraft's systems performance, like: Right engine turbine shell vibration [mm/s], Oil pressure in the RE [kg/cm2] and Air temperature in RE inlet [°C] which are used by the aircraft technical staff to monitor the aircraft performance, therefore, they update the aircraft Maintenance Logbook and schedule the maintenance activates for the aircraft. These parameters when they are formed in charts and tables by the Dvina-M software, they become useful and easy to be analysed by the technical staff, which represents the benefit of Dvina-M software.

The remained parameters represent the aircraft behavior and attitude during flight and the effects of the pilot use of aircraft flight control systems, these parameters include angle of pitch [deg], angle of heading [deg], Indicated air speed, Aileron angle [deg], Landing gear raised and etc. These parameters of no use when they are represented in tables and charts which are produced by Dvina-M software, that's where the instructors pilots and their students can't trace and analysis the flight session. The instructors pilot need data to be represented in form of aircraft model that simulate the behavior and motion of aircraft and the use of flight controls, this aircraft model must be equipped with cockpit instruments which represents the parameters values as indicators of the

aircraft's behavior and motion. In addition, the pilot's instructors need to analyse the flight path to measure the pilot's performance during each part of the flight session (Takeoff, departure, cruise, maneuvers, aerobatics, approach, landing, etc).

As a fact of practice, the data which is recorded by the Тестер У3-ЛК flight data recorder is used for aircraft technical tasks, but have no use in pilot's training or measures of pilot performance during flight. This represents the insufficiency of Dvina-M software.

3.3. Microsoft Flight Simulator as training aid

Flight Simulator 2004 - A Century of Flight, also known as FS9 or FS2004, was shipped with several historical aircraft such as the Wright Flyer, Ford Tri-Motor, and the Douglas DC-3 to commemorate the 100th anniversary of the Wright Brothers' first flight. The program included an improved weather engine that provided true three-dimensional clouds and true localized weather conditions for the first time. The engine also allowed users to download weather information from actual weather stations, allowing the simulator to synchronize the weather with the real world. In addition, the simulator includes ATC communications, GPS equipment, interactive virtual cockpits, and other features. These features are designed to make simulated flight realistic, easy, entertaining, and can be used to learn real-world skills more efficiently and effectively with capability of using instructor's station that makes it an excellent training aid. A part of the Microsoft Flight Simulator features includes:

1. Creating, saving and selecting a flight: which will be saved in "C:\Documents and Settings*myname*\My Documents\Flight Simulator Files" in MS Windows XP and later.

2. Weather features make it easy to create cloud layers, pea-soup fog, crosswinds, and other challenges.

3. Enhanced Flight Analysis feature that acts much like a flight data recorder in a real-world airplane. It automatically tracks altitude, heading, speed, and other information. The analysis includes how well ATC's instructions are followed or tracked the localizer and glide slope on an ILS, with feature of play back the flight overlaid on a map.

4. Map view shows the location of navigation aids, low and high altitude airways, intersections, and aircraft track over the ground. The Map view also serves as an airport/facility directory, it produces information about navigational aids, radio frequencies, airport elevations, and other relevant navigational details, right from the map. GPS is the source of Map view information.

5. Multiple views feature enhanced the simulation capabilities by including five main view modes which can be monitored in five separate windows. The five views are Cockpit view, Virtual Cockpit view, Tower view, Spot plane view and Top-down view.

6. Video recorder feature acts as the flight data recorder on an airplane. It periodically samples airspeed, altitude, heading, and other flight data. It saves the information in a small file but it does not create an AVI or MPEG video file. During play back it runs the captured data back through the simulation engine.

7. The autopilot feature: which can handle some of the flying load while the pilot is learning a new skill.

8. Variety of downloadable aircrafts: where in this simulation program the IRIS Fulcrum aircraft is downloaded and installed to the flight simulator.

3.4. Flight Simulator Universal Inter-Processes Communications (FSUIPC)

FSUIPC is a module which is designed to allow external programs to communicate with and perhaps control Microsoft Flight Simulator by access many of the Flight Simulator internal variables from simple data like pitch, roll, altitude and heading right through to weather and acceleration values. Flight Simulator panels (including their gauges), aircraft, scenery and other graphics that are mostly not correctable or influenced in any way by FSUIPC. In other words, FSUIPC can only help external applications talk to FS. FSUIPC.dll is a module for FS2000, FS2002 or FS2004 and it should be placed into the Flight Simulator modules folder and become part of the FS process and runs inside FS and in the same memory space of FS. The FSUIPC module reserves a 65535 byte block of memory and then populates a part of this memory with various pieces of data which are extracted from FS. External applications then can read and write data in this blocks of memory for their own use.

Each piece of FS data is placed in the memory block at a certain location is referred to as "offset", while all the offsets numbers (0000_H-$FFFF_H$) are listed in the FSUIPC_SDK with their use and sizes in bytes and whether they can be read from or written to. These offsets were originally true offsets into a specific global data area within FS but in FS2004 and later, they are treated as identifiers to specific variables and they are translated within FSUIPC. However, still address data with contiguous offsets in blocks can be addressed, and FSUIPC breaks it down if it needs to. Table (3.1) illustrates the offsets that were used to simulate the aircraft motion from FDR recorded data.

To develop a Visual Basic program to communicate and control the Microsoft Flight Simulator the FSUIPC.bas module is inserted into the VB project, therefore, it provide the interface between the VB project and the

FSUIPC module in the Flight Simulator. The interface is provided through calling the appropriate library routines in the VB code. The communication goes through the following stages:

1. Call FSUIPC_Initialization() to initialize important variables at the beginning of the program.
2. Call FSUIPC_Open() to connect with FSUIPC if it's available in the flight simulator and the simulator is ready.
3. Call a series of FSUIPC_Read and/or FSUIPC_Write and/or FSUIPC_WriteS
4. Call FSUIPC_Process() to actually get/set the data from/to FS.
5. Steps 3 and 4 repeated as necessary.
6. At program termination, call FSUIPC_Close().

Table (3.1): *Offsets used by the FDR data simulation program.*

Offset	Size	Use	FS Read	FS Write
0262	2	Pause control (write 1 to pause, 0 to un-pause).	N/A	Ok-SimC
0570	8	Altitude, in meters and fractional meters. The units are in the high 32-bit integer (at 0574) and the fractional part is in the low 32-bit integer (at 0570). [Can be written to move aircraft]	Ok-SimC	Ok-SimC
0578	4	Pitch, *360/(65536*65536) for degrees. 0=level, –ve=pitch up, +ve=pitch down	Ok-SimC	Ok-SimC
057C	4	Bank, *360/(65536*65536) for degrees. 0=level, –ve=bank right, +ve=bank left	Ok-SimC	Ok-SimC
0580	4	Heading, *360/(65536*65536) for degrees TRUE	Ok-SimC	Ok-SimC
07BC	4	Autopilot Master switch	Ok-SimC	Ok-SimC
07DC	4	Autopilot airspeed hold	Ok-SimC	Ok-SimC
07E2	4	Autopilot airspeed value, in knots	Ok-SimC	Ok-SimC
0810	4	Auto-throttle Arm	Ok-SimC	Ok-SimC
088C	2	Engine 1 Throttle lever, –4096 to +16384 [Programs controlling throttle directly from user inputs should write to 089A instead if the input should be disconnectable via offset 310A (e.g. for auto-throttle management)]	Ok-SimC	Ok-SimC
0924	2	Engine 2 Throttle lever, –4096 to +16384 [Programs controlling throttle directly from user inputs should write to 0932 instead if the input should be disconnectable via offset 310A	Ok-SimC	Ok-SimC

		(e.g. for auto-throttle management)]		
0BB2	2	Elevator control input: –16383 to +16383	Ok-SimC	Ok-SimC
0BB6	2	Aileron control input: –16383 to +16383	Ok-SimC	Ok-SimC
0BBA	2	Rudder control input: –16383 to +16383	Ok-SimC	Ok-SimC
0BDC	4	Flaps control, 0=up, 16383=full. The —notches‖ for different aircraft are spaced equally across this range: calculate the increment by 16383/(number of positions-1), ignoring fractions. See also offset 3BFA below.	Ok-SimC	Ok-SimE
0BE8	4	Gear control: 0=Up, 16383=Down	Ok-SimC	Ok-SimC
3364	1	FS2004 —Ready to Fly‖ indicator. This is non-zero when FS is loading, or reloading a flight or aircraft or scenery, and becomes zero when flight mode is enabled (even if the simulator is paused or in Slew mode).	Ok-SimE	N/A

In stage three, in each call the offset number and size in bytes are sent within the function call, with pointer to the area to receive the result of read function or the area containing the data to be written by the write function. The received data from the read function or the data to be written by the write function is the desired parameter's value. This parameter's values are converted according to the attached equation with the offset number in the FSUIPC_SDK to the correct parameter value and unit. Therefore, in order to simulate the flight session using the parameters values recorded in the FDR. The extracted parameters values from the flight data recorder are compared to those used by the flight simulator aircraft and reformed according to their equations to meet the specifications of the flight simulator's aircraft parameters.

Considering Pitch angle parameter as an example, the parameter value needs to be sent from FDR data where it is measured as an angle in degrees to the flight simulator parameters which represents the Pitch angle in 4-bytes integer number. Referring to the offset number 0588_H in the FSUIPC_SDK, to get the aircraft Pitch angle from the flight simulator, the

long integer number in the offset 0588$_H$ multiplied by 360/(65536*65536) to be converted to the aircraft Pitch angle in degrees. The positive value represent Pitch up (climb) and negative value represent Pitch down (decline), therefore to send the FDR's Pitch angle to the flight simulator, the Pitch value multiplied by (65536*65536) /360 (the reverse of read equation) to be converted to the flight simulator representation of Pitch angle.

3.5. Simulation Program code

The simulation software is programmed in Visual Basic with the use of FSUIPC.bas module which is inserted into the project, and the user control (FSControl) which is used to communicate through the FSUIPC module to the flight simulator. The main program code is used to manage the FDR data files and to communicate with flight simulator by managed timing, parameters value, airports, flight simulation sessions, etc.

3.5.1. The user control "FSControl"

The FSControl uses basically two functions to communicate with the flight simulator, the setValue () function to send data to flight simulator and GetInfo () function to parameters values from flight simulator. The benefit of using FSControl user control with the FSUIPC module is that it uses enumeration for the flight parameters that need to be read/written from/to flight simulator and combine them with the functions setValue () and GetInfo (), thus the flight parameters names are declared as enumeration members and are used as parameters for the two functions. This provides meaningful flight parameters names which are easier to program with than their offsets, sizes and values. Figure (3.3-a) shows the declaration of the enumeration "FSAttrib" and the flight parameters as enumeration members, while Figure (3.3-b) shows the declaration of the GetInfo () function with the enumeration "FSAttrib" as data type of the function parameter "Attrib", the body of the function shows the use of enumeration

members (Flight parameter name) to communicate through FSUIPC module's functions to get flight parameter value from the flight simulator. Basing on the above explanation of enumeration with the functions setValue () and GetInfo (), all the flight parameters which are used in the simulation program and detailed in Table (3.1) above, were coded as enumeration member and inserted into the FSControl functions by inserting the flight parameters conversion equations, the offsets, sizes and FSUIPC read/write and process functions. Table (3.2) illustrates the Visual Basic code for those flight parameters used with the GetInfo () function, while Table (3.3) illustrates the Visual Basic code for those flight parameters used with the setValue () function.

| (a) | (b) |

Figure (3.3): *FSControl enumeration and GetInfo() function.*

Table (3.2): *flight parameters used with the GetInfo () function.*

No.	Flight parameter	Enum. member name	VB Code in GetInfo () function

1	Ready to Fly	fsReady	If Attrib = fsReady Then
			If FSUIPC_Read(&H3364, 1,VarPtr(Tmp),dwResult)Then
			    If FSUIPC_Process(dwResult) Then
			      GetInfo = Tmp
			    End If
			End If
			End If

2	Altitude	fsAltitude	If Attrib = fsAltitude Then
			If FSUIPC_Read(&H574,4, VarPtr(Tmp), dwResult) Then
			    If FSUIPC_Process(dwResult) Then
			      GetInfo = Round(Tmp * 3.3, 0)
			    End If
			End If
			End If

Table (3.3): *flight parameters used with the setValue () function.*

No.	Flight parameter	Enum. member name	VB Code in setValue () function

1	Autopilot airspeed value, in knots	fsASpeed	If setWhat = fsASpeed Then
			    x = settingNumber * 0.53996
			    FSUIPC_Write &H7E2, 2, VarPtr(x), dwResult
			End If

2	Elevator control input	fsElevator	If setWhat = fsElevator Then
			    x = Round((16383 * settingNumber / 100), 0)
			    FSUIPC_Write &HBB2, 2, VarPtr(x), dwResult
			End If

3		fsAileron	If setWhat = fsAileron Then

	Aileron control input		x = Round((16383 * settingNumber / 100), 0) FSUIPC_Write &HBB6, 2, VarPtr(x), dwResult End If

4	Engine 1 Throttle lever	fsThrottle1	If setWhat = fsThrottle1 Then
			    x = 16383 * (settingNumber / 100)
			    FSUIPC_Write &H88C, 2, VarPtr(x), dwResult
			End If

5	Engine 2 Throttle lever	fsThrottle2	If setWhat = fsThrottle2 Then
			    x = 16383 * (settingNumber / 100)
			    FSUIPC_Write &H924, 2, VarPtr(x), dwResult
			End If

6	Flaps control	fsFlaps	If setWhat = fsFlaps Then
			    x = settingNumber * 16383
			    FSUIPC_Write &HBDC, 4, VarPtr(x), dwResult
			End If

7	Landing Gear control	fsGear	If setWhat = fsGear Then
			  If settingNumber = 0 Then
			    x = 16383
			  End If
			  If settingNumber = 1 Then
			    x = 0
			  End If
			    FSUIPC_Write &HBE8, 4, VarPtr(x), dwResult
			End If

8	Parking brake	fsBrake	If setWhat = fsBrake Then '0BC8
			    x = 16383 * (settingNumber / 100)
			    FSUIPC_Write &HBC8, 2, VarPtr(x), dwResult
			End If

9	Autopilot Master switch	fsAP	If setWhat = fsAP Then
			    x = settingNumber
			    FSUIPC_Write &H7BC, 4, VarPtr(x), dwResult
			End If

10	Autothrottle Arm	fsAThrottle	If setWhat = fsAThrottle Then x = settingNumber

			FSUIPC_Write &H810, 4, VarPtr(x), dwResult End If

11	Autopilot airspeed hold	fsASpeedH	If setWhat = fsASpeedH Then
			    x = settingNumber
			    FSUIPC_Write &H7DC, 4, VarPtr(x), dwResult
			End If

12	Heading	fsHeading2	If setWhat = fsHeading2 Then
			    x = (settingNumber / 360) * 65536 * 65536
			    FSUIPC_Write &H580, 4, VarPtr(x), dwResult
			End If

13	Altitude	fsAlt	If setWhat = fsAlt Then
			    x = settingNumber
			    FSUIPC_Write &H574, 4, VarPtr(x), dwResult
			End If

14	Bank angle	fsBank2	If setWhat = fsBank2 Then
			    x = (65536 / 360) * 65536 * (-settingNumber)
			    FSUIPC_Write &H57C, 4, VarPtr(x), dwResult
			End If

15	Pitch angle	fsPitch2	If setWhat = fsPitch2 Then
			    x = (65536 / 360) * 65536 * (-settingNumber)
			    FSUIPC_Write &H578, 4, VarPtr(x), dwResult
			End If

16	Pause control	fsPause	If setWhat = fsPause Then
			    FSUIPC_Write &H262, 2,
			    VarPtr(settingNumber), dwResult
			End If

3.5.2. Real Flight Simulation program code

The Real Flight Simulation program code is written in Visual Basic after insertion of the FSUIPC module and the user Control FSControl, Figure(3.4) shows the VB project Form. The Form has two menus File and Help. The File menu consists of three submenus which are New Flight, Open flight and Exit, the Help menu consists of one submenu About. The

Form also includes thirteen TextBox which are used to read the FDR parameters values from the FDR data files, the TexBox preceded by Labels to indicate the parameters names of the TexBoxs. The Form includes four Timers, CommonDialog and picture of the aircraft that is used in the simulation sessions.
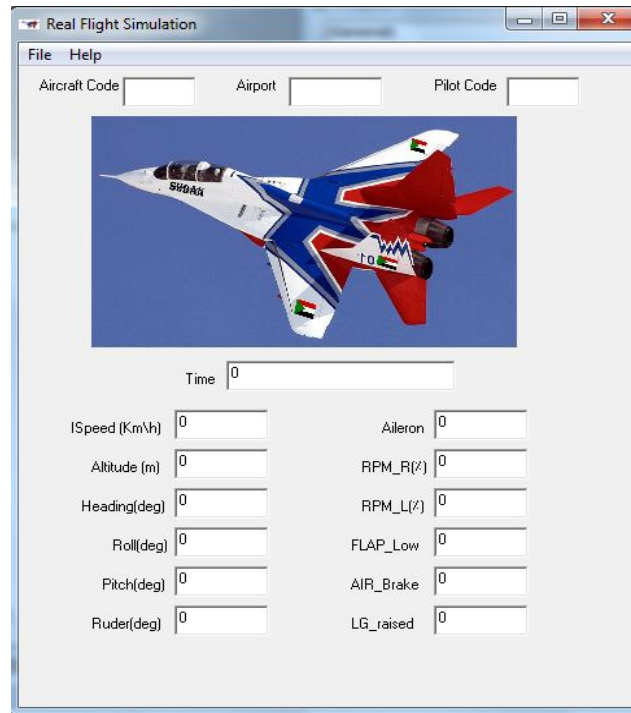
Figure (3.4): *The VB project Form*.

The main program code divided into four parts start/exit program, new flight files preparation, open existed flight and flight simulation.

3.5.2.1. Start/exit program

Starting the program is controlled by running the program which loads the Form "Form1" without any tasks performed during loading, however exiting the program is done through submenu "Exit" in the main menu File or the window close icon. Figure (3.5) shows the main flowchart of the Real Flight Simulation program.
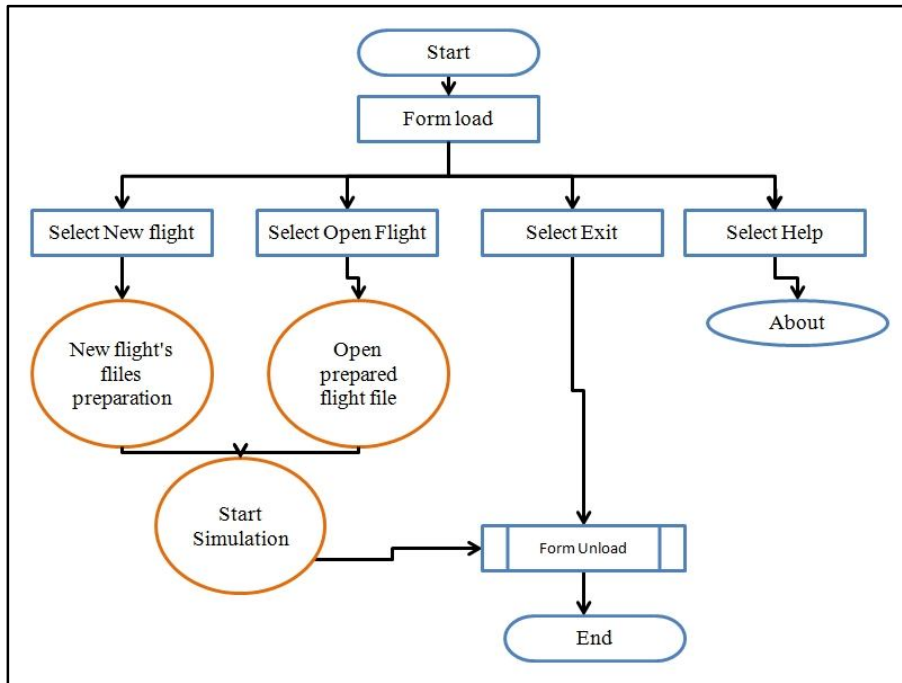
Figure (3.5): *Real Flight Simulation program main flowchart*

Exiting the program executes the Form_Unload code to perform the following:

1. Turnoff the flight simulator's autopilot, autothrottle and autopilot hold speed

   FSControl1.setValue fsAP, 0

   FSControl1.setValue fsAThrottle, 0

   FSControl1.setValue fsASpeedH, 0

2. Turnoff the program's four Timers

   Timer1.Interval = 0

   Timer2.Interval = 0

   Timer3.Interval = 0

   Timer4.Interval = 0

3. Close all FDR data files

   Close #iFileNoR1

   Close #iFileNoR2

   Close #iFileNoW

4. Close the FSControl and the FSUIPC session

   FSControl1.FSStop

FSUIPC_Close

5. Close the Microsoft Flight Simulator 2004 session

  Dim FSClose As Long

  FSClose = FindWindow(vbNullString, "Microsoft Flight Simulator 2004 - A Century of Flight")

  PostMessage FSClose, WM_CLOSE, 0&, 0&

3.5.2.2. New flight files preparation

The New flight files preparation stage flowchart is shown in Figure (3.6), where the stage starts by choosing the submenu "New flight" from the main menu File, which is used to read the FDR flight parameters values that are required for simulation from two text files and rearrange them in a new text file. The process goes through the following steps:

1. Open the CommonDialog window and chose a text file

  CommonDialog1.Filter = "Text Files|*.txt"

  CommonDialog1.ShowOpen

  If CommonDialog1.FileName = "" Then Exit Sub

2. Assign a free file number for the opened file, then read one line and split the text in string array every "\" character

  iFileNoR1 = FreeFile

  Close #iFileNoR1

  Open CommonDialog1.FileName For Input As #iFileNoR1

  Input #iFileNoR1, TextFileNo1

  strSpilit() = Split(TextFileNo1, "\")

3. Validate the file through ensures that there is "Dvina-M.wrk" software name

  If EOF(iFileNoR1) Or strSpilit(1) <> "Dvina-M.wrk" Or strSpilit(2) _

        <> "MIG_29" Then

   MsgBox ("ERROR...............File unsupported" _

      + CommonDialog1.FileName)

   Close #iFileNoR1

   Exit Sub
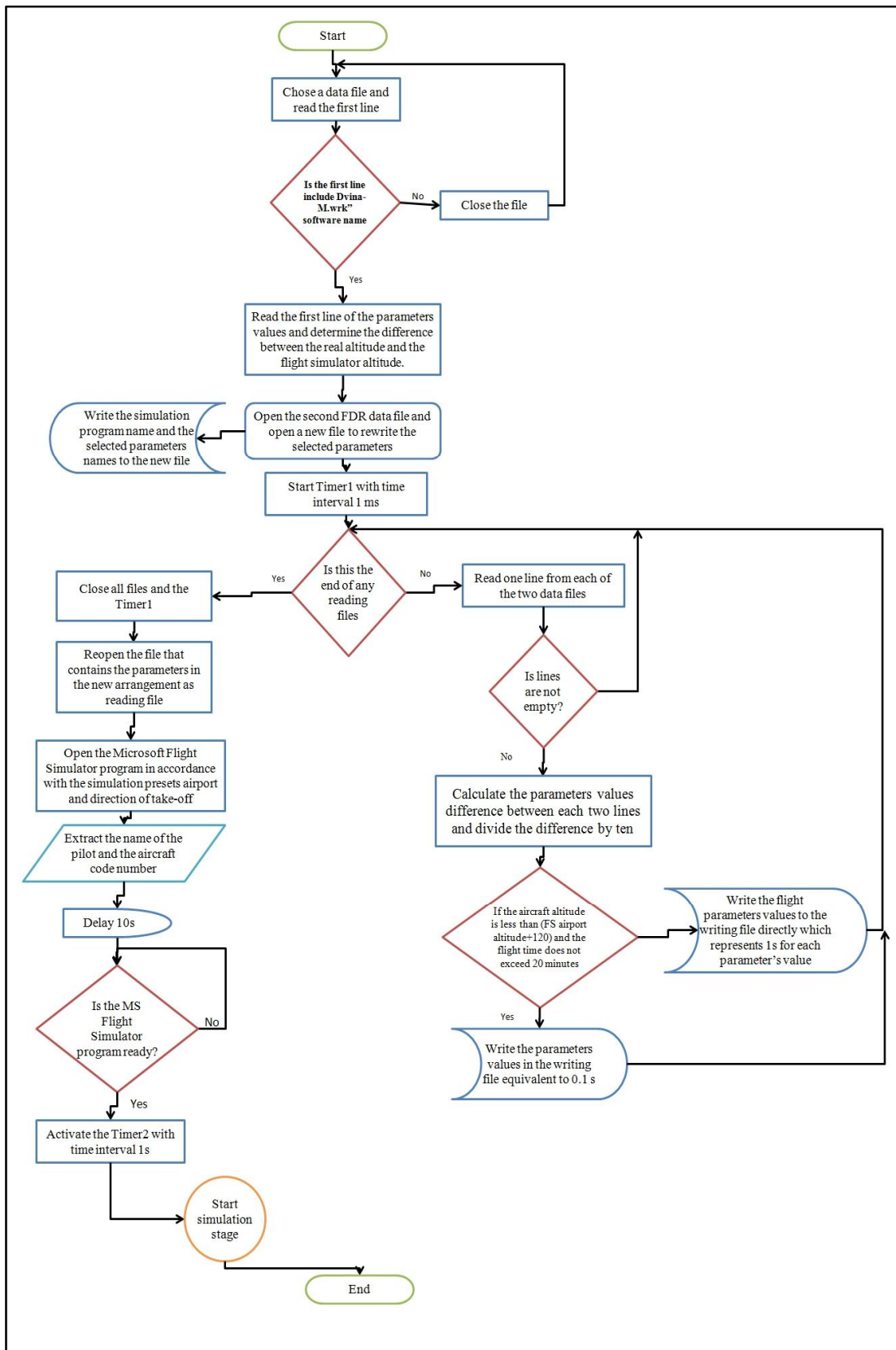
Figure (3.6): *New flight files preparation stage flowchart*

4. In case of correct file and data, read four lines undesirable down to read the first line of the parameters values and split it in string array every "|" character. Then extract the airport altitude (where the flight took off from) in order to determine the difference between the real altitude and the flight simulator altitude.

```
Input #iFileNoR1, TextFileNo1
Input #iFileNoR1, TextFileNo1
Input #iFileNoR1, TextFileNo1
 Input #iFileNoR1, TextFileNo1
 Input #iFileNoR1, TextFileNo1
 strSpilit1() = Split(TextFileNo1, "|")
 FDRAirportAlt = strSpilit1(13)
 Select Case FDRAirportAlt

         Case Is < 50
            AltDiff = 45 - FDRAirportAlt
         Case Is < 160
            AltDiff = 260 - FDRAirportAlt
         Case Is < 340
            AltDiff = 404 - FDRAirportAlt
         Case Is < 400
            AltDiff = 239 - FDRAirportAlt
         Case Is < 480
            AltDiff = 577 - FDRAirportAlt
         Case Is < 590
            AltDiff = 658 - FDRAirportAlt
         Case Is < 790
            AltDiff = 735 - FDRAirportAlt
         Case Else
            MsgBox ("Did not find an airport")
            GoTo E:
      End Select
```

5. Open the second FDR data file after assigning a free file number and open a new file to rewrite the selected parameters in a new order. Write the simulation program name and the selected parameters names to the new file.

```
iFileNoW = FreeFile
Close #iFileNoW
Open Replace$(CommonDialog1.FileName, "1.txt", "Simu.txt") For _
```

Output As #iFileNoW

Me.MousePointer = vbHourglass

Print #iFileNoW, "Flight Simulation Program File"

Print #iFileNoW, " Time | Speed |Altitude| Hdg | Roll | Pitch |" _
        & "Rudder |Aileron| N2R | N2L |Fl|LG|"

6. Start Timer1 with time interval 1 ms. This timer used to read the selected parameters values from the two opened FDR data files, then write these values to the new file opened for writing.

Timer1.Interval = 1

7. Use Timer1 to read one line from each of the two data files and split the parameters values in two string arrays and making sure that the lines are not empty.

If Not EOF(iFileNoR1) Or EOF(iFileNoR2) Then
        Input #iFileNoR1, TextFileNo1
        strSpilit3() = Split(TextFileNo1, "|")
        Input #iFileNoR2, TextFileNo2
        strSpilit2() = Split(TextFileNo2, "|")
        If TextFileNo1 = vbNullString Or TextFileNo2 = vbNullString _
                                Then
            On Error GoTo E:
        End If

8. Calculate the parameters values difference between each two lines and divide the difference by ten.

SpeedRate = Round((strSpilit3(11) - strSpilit1(11)) / 10, 1)

AltRate = Round((strSpilit3(13) - strSpilit1(13)) / 10, 1)

HeadRate = Round((strSpilit3(18) - strSpilit1(18)) / 10, 2)

RollRate = Round((strSpilit3(19) - strSpilit1(19)) / 10, 2)

PitchRate = Round((strSpilit3(20) - strSpilit1(20)) / 10, 2)

RudderRate = Round(((strSpilit3(24) - strSpilit1(24)) / 10), 2)

AlrRate = Round(((strSpilit3(28) - strSpilit1(28)) / 10), 2)

N2RRate = Round((strSpilit3(32) - strSpilit1(32)) / 10, 1)

N2LRate = Round((strSpilit3(33) - strSpilit1(33)) / 10, 1)

9. If the aircraft altitude is less than (FDRAirportAlt + AltDiff + 120) and the flight time does not exceed 20 minutes, write the flight parameters values to the writing file directly which represents 1s for each parameter's value, while transferring the modern values of the parameters to the second string array.

```
If  strSpilit3(13)  <=  (FDRAirportAlt  +  AltDiff  +  120)  And  _
TimeValue(Left$(strSpilit3(0), 8)) < TimeValue("00:20:00") Then
        Print   #iFileNoW,   RTrim$(strSpilit3(0))   &   "|"   &   _
        Format(strSpilit3(11),     "0000.00")     &     "|"     &     _
        Format(strSpilit3(13),     "00000.00")     &     "|"     &     _
        Format(strSpilit3(18), "000.00")&"|" & Format(strSpilit3(19),_
        "000.00") & "|" & Format(strSpilit3(20), "000.00") & "|" & _
        Format(strSpilit3(24),"000.00")& "|" &Format(strSpilit3(28), _
        "000.00") & "|" & Format(strSpilit3(32), "000.00") & "|" &_
        Format(strSpilit3(33),       "000.00")       &       "|"       &
        RTrim$(strSpilit2(23)) & "|" & RTrim$(strSpilit2(25)) & "|"_
            strSpilit1(11) = strSpilit3(11)
            strSpilit1(13) = strSpilit3(13)
            strSpilit1(18) = strSpilit3(18)
            strSpilit1(19) = strSpilit3(19)
            strSpilit1(20) = strSpilit3(20)
            strSpilit1(24) = strSpilit3(24)
            strSpilit1(28) = strSpilit3(28)
            strSpilit1(32) = strSpilit3(32)
            strSpilit1(33) = strSpilit3(33)
```

10. If the aircraft altitude exceeds (FDRAirportAlt + AltDiff + 120) or the flight time exceeded 20 minutes, write the parameters values in the writing file equivalent to 0.1 s. Reduce the time period for the parameters values to 0.1s increases the accuracy of simulation ten times.

```
For x = 1 To 10
        WriteLine = ""
        strSpilit3(0) = Left$(strSpilit3(0), 8)
```

```
                    strSpilit3(0) = strSpilit3(0) & Format(x * 0.099, ".00")
                    strSpilit1(11) = strSpilit1(11) + SpeedRate
                    strSpilit3(1) = Format(strSpilit1(11), "0000.00")
                    strSpilit1(13) = strSpilit1(13) + AltRate
                    strSpilit3(2) = Format(strSpilit1(13), "00000.00")
                    strSpilit1(18) = strSpilit1(18) + HeadRate
                    strSpilit3(3) = Format(strSpilit1(18), "000.00")
                    strSpilit1(19) = strSpilit1(19) + RollRate
                    strSpilit3(4) = Format(strSpilit1(19), "000.00")
                    strSpilit1(20) = strSpilit1(20) + PitchRate
                    strSpilit3(5) = Format(strSpilit1(20), "000.00")
                    strSpilit1(24) = strSpilit1(24) + RudderRate
                    strSpilit3(6) = Format(strSpilit1(24), "000.00")
                    strSpilit1(28) = strSpilit1(28) + AlrRate
                    strSpilit3(7) = Format(strSpilit1(28), "000.00")
                    strSpilit1(32) = strSpilit1(32) + N2RRate
                    strSpilit3(8) = Format(strSpilit1(32), "000.00")
                    strSpilit1(33) = strSpilit1(33) + N2LRate
                    strSpilit3(9) = Format(strSpilit1(33), "000.00")
                    strSpilit3(10) = RTrim$(strSpilit2(23))
                    strSpilit3(11) = RTrim$(strSpilit2(25))
                      For y = 0 To 11
                        WriteLine = WriteLine & strSpilit3(y) & "|"
                      Next y
                     Print #iFileNoW, WriteLine
          Next x
```

11. At the end of any reading files, close all files and the Timer1, then reopen the file that contains the parameters in the new arrangement as reading file and read three lines undesirable.

```
          Close #iFileNoW
          Close #iFileNoR1
          Close #iFileNoR2
          Timer1.Interval = 0
          Me.MousePointer = vbNormal
```

```
MsgBox ("Write File " +Replace$(CommonDialog1.FileName, "1.txt",_
                            "Simu.txt") + " closed")
Open Replace$(CommonDialog1.FileName, "1.txt", "Simu.txt") For_
                            Input As #iFileNoR1
Input #iFileNoR1, TextFileNo1
Input #iFileNoR1, TextFileNo1
Input #iFileNoR1, TextFileNo1
```

12. Basing on the airport altitude in paragraph 4 above, open the Microsoft Flight Simulator program in accordance with the simulation presets airport and direction of take-off, then extract the name of the pilot and the aircraft code number and suspend the execution for 10 seconds to verify that the simulation program ready for action.

```
If FDRAirportAlt < 50 Then
        FSstart = ShellExecute(GetDesktopWindow(), "Open", _
        "C:\Users\Waleed\Documents\Flight Simulator Files\ _
        PortSudan.flt ", "", "C:\", 1)
        Airport = "Portsudan"
ElseIf FDRAirportAlt > 60 And FDRAirportAlt < 160 Then
        FSstart = ShellExecute(GetDesktopWindow(), "Open", _
        "C:\Users\Waleed\Documents\Flight Simulator Files\ _
        Merowe.flt", "", "C:\", 1)
        Airport = "Merowe"
ElseIf FDRAirportAlt > 230 And FDRAirportAlt < 340 Then
        FSstart = ShellExecute(GetDesktopWindow(), "Open", _
         "C:\Users\Waleed\Documents\Flight Simulator Files\WADI-_
        H38.flt", "", "C:\", 1)
        Airport = "Wadi Saydna"
ElseIf FDRAirportAlt > 360 And FDRAirportAlt < 400 Then
         FSstart = ShellExecute(GetDesktopWindow(), "Open", _
         "C:\Users\Waleed\Documents\Flight Simulator Files\ _
         Dongola.flt", "", "C:\", 1)
         Airport = "Dongola"
```

```vbnet
ElseIf FDRAirportAlt > 420 And FDRAirportAlt < 480 Then
            FSstart = ShellExecute(GetDesktopWindow(), "Open", _
            "C:\Users\Waleed\Documents\Flight Simulator Files\El _
            Obaid.flt", "", "C:\", 1)
            Airport = "El Obaid"
ElseIf FDRAirportAlt > 500 And FDRAirportAlt < 590 Then
            FSstart = ShellExecute(GetDesktopWindow(), "Open", _
            "C:\Users\Waleed\Documents\Flight Simulator Files\ _
            Nyala.flt", "", "C:\", 1)
            Airport = "Nyala"
ElseIf FDRAirportAlt > 620 And FDRAirportAlt < 790 Then
            FSstart = ShellExecute(GetDesktopWindow(), "Open", _
            "C:\Users\Waleed\Documents\Flight Simulator Files\ _
            Alfashir.flt", "", "C:\", 1)
            Airport = "Alfashir"
Else
            MsgBox ("Did not find an airport")
            GoTo z:
End If
ACCode = Mid$(CommonDialog1.FileName,InStr(CommonDialog1. _
            FileName, "6"), 3)
PilotName=Mid$(CommonDialog1.FileName,InStr(CommonDialog1._
            FileName, "6") + 4, 4)
DoEvents
Sleep (10000)
FSControl1.FSstart
Do While FSstart > 31 And FSControl1.GetInfo(fsReady) <> 0
            DoEvents
Loop
```

13. Extract the MS Flight Simulator airport altitude and then start the simulation by activating the Timer2 with time interval 1s.

```vbnet
FSAirportAlt = Round((FSControl1.GetInfo(fsAltitude) / 3.3), 0)
Timer2.Interval = 1000
```

3.5.2.3. Open existed flight

Open an existed flight provides the ability of using an FDR data file of a previous simulation session, the FDR data file will be modified and the parameters rearranged before the previous session and the data file name will be extended by "Simu" as annotation for the modified FDR data files. Figure (3.7) shows the flowchart of the Open and existed flight stage.

Open an existed flight goes through the following steps after choosing the submenu "Open flight" from the main menu File:

1. Open the CommonDialog window and chose a text file

> CommonDialog1.Filter = "Text Files|*.txt"
>
> CommonDialog1.ShowOpen
>
> If CommonDialog1.FileName = "" Then Exit Sub

2. Assign a free file number for the opened file then read and validate the file through ensuring that the first line is the simulation program name

> Open CommonDialog1.FileName For Input As #iFileNoR1
>
> Input #iFileNoR1, TextFileNo1
>
> If EOF(iFileNoR1) Or TextFileNo1 <> "Flight Simulation Program _
>
> > File" Then
> >
> > MsgBox ("ERROR................File unsupported " + TextFileNo1)
> >
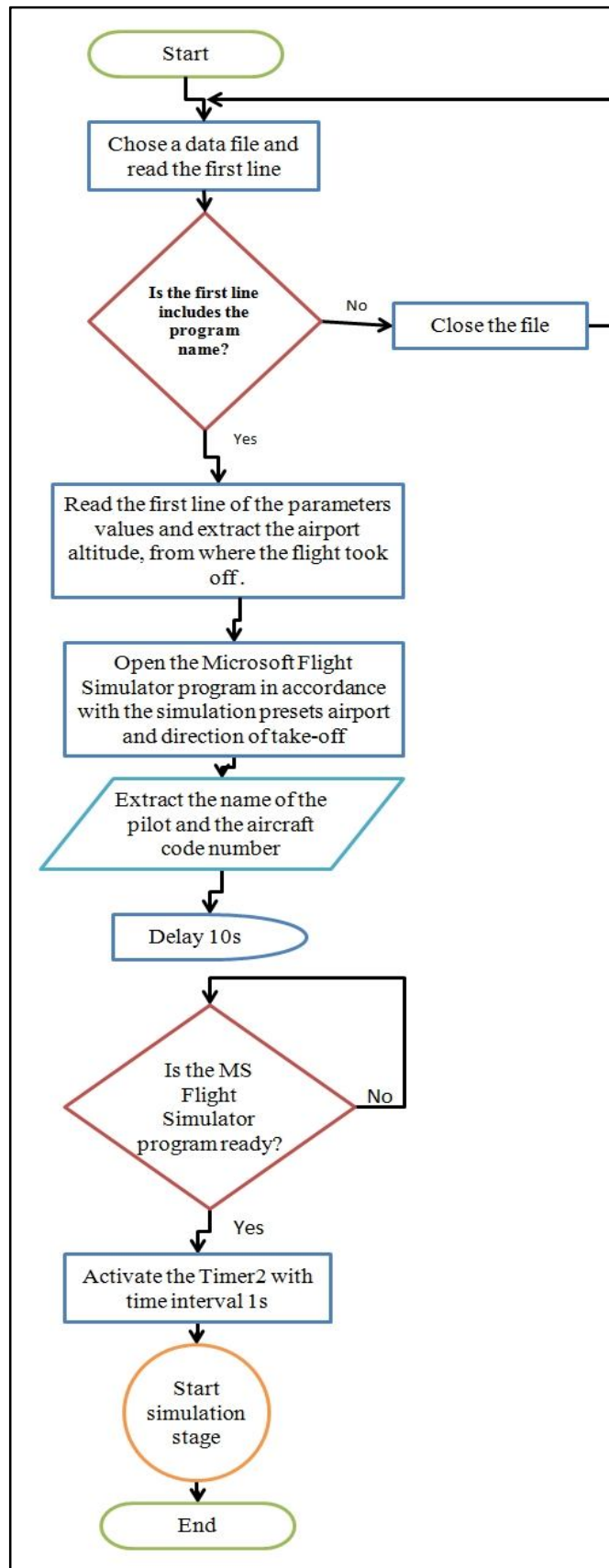> > Close #iFileNoR1
> >
> > Exit Sub

Figure (3.7): *Open and existed flight stage's flowchart*

3. In case of correct file and data, read two undesirable lines down to read the first line of the parameters values, and split it in string array every "|" character. Then extract the airport altitude, where the flight took off from.

```
Input #iFileNoR1, TextFileNo1
Input #iFileNoR1, TextFileNo1
strSpilit() = Split(TextFileNo1, "|")
FDRAirportAlt = strSpilit(2)
```

4. Based on the airport altitude, open the Microsoft Flight Simulator program in accordance with the simulation presets airport and direction of take-off, then extract the name of the pilot and the aircraft code number and suspend the execution for 10 seconds to verify that the simulation program ready for action. The process done by same code as in (**3.5.2.2.** paragraph 12) above.

5. Extract the MS Flight Simulator airport altitude and then start the simulation by activates the Timer2 with time interval 1s. The process done by the same code as in (**3.5.2.2.** paragraph 13) above.

3.5.2.4. Flight simulation

Since the simulation is the primary mission of the program and the most important step where it followed by the process of analysis, therefore the simulation phase has been divided into three subsidiary stages, in accordance with the phases of the flight session, with accuracy intense, led to increase the size of parameters values sent to the simulation program ten times the true extracted values from the FDR. The three subsidiary stages are Before flight preparation/take-off, Flying and Landing. Figure (3.8) shows the simulation flowchart.

3.5.2.4.1. Before flight preparation and takeoff

This is the preparatory stage where the before flight tests of the aircraft took place, taxis up toward the runway and standby for take-off.
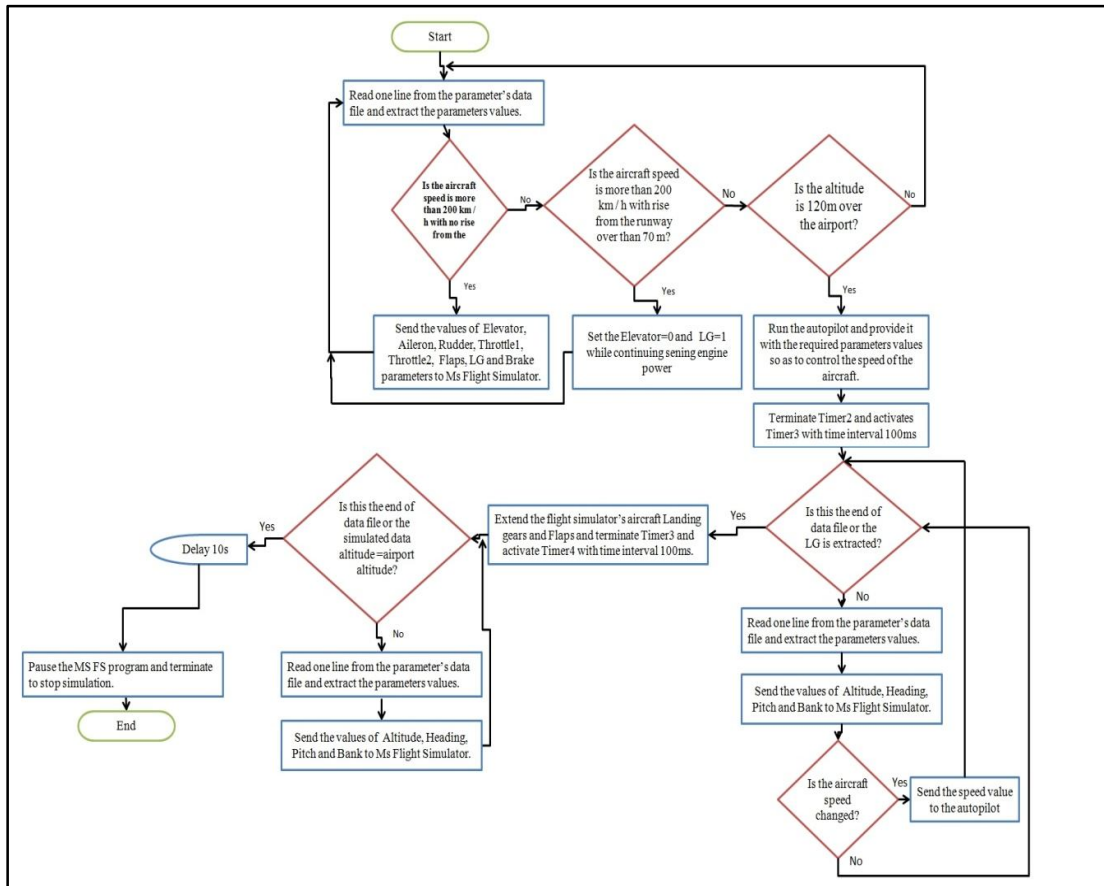
Figure (3.8): *Flight simulation stage's flowchart*

This stage often extends between 10 to 15 minutes and is followed by the take-off stage. The simulation starts when one of the previous program parts (New flight files preparation or Open existed flight) activates Timer2 with time interval 1s. The following code is executed within each cycle of Timer2:

1. Read one line from the parameter's data file, then split the text line in a string array and extract the parameters values.

```
If Not EOF(iFileNoR1) Then
        Input #iFileNoR1, TextFileNo1
        strSpilit() = Split(TextFileNo1, "|")
        F_Time = strSpilit(0)
        Speed = strSpilit(1)
        Altitude = strSpilit(2)
        Heading = strSpilit(3)
```

```
                Rudder = strSpilit(6)
                Aileron = strSpilit(7)
                N2R = strSpilit(8)
                N2L = strSpilit(9)
                Flap = strSpilit(10)
                LG = strSpilit(11)
```

2. Check out the end of aircraft test period, this is done if the aircraft speed is more than 200 km / h with no rise from the runway. Then send the values of aircraft controls parameters to the flight simulator program.

```
    If Speed > 200 And Round((FSControl1.GetInfo(fsAltitude) / 3.3), 0) _
            < (FSAirportAlt + 50) Then
            FSControl1.setValue fsElevator, 45
            FSControl1.setValue fsAileron, Aileron
            FSControl1.setValue fsRudder, Rudder
            FSControl1.setValue fsThrottle1, N2R
            FSControl1.setValue fsThrottle2, N2L
            FSControl1.setValue fsFlaps, Flap
            FSControl1.setValue fsGear, LG
            FSControl1.setValue fsBrake, 0
```

3. Check the end of takeoff period, this is done if the aircraft speed is more than 200 km / h with rise from the runway over than 70 m, then retract the Landing gears and Flaps while continuing sening engine power.

```
    ElseIf Speed > 200 And Round((FSControl1.GetInfo(fsAltitude) / 3.3),_
            0) > (FSAirportAlt + 70) Then
            FSControl1.setValue fsElevator, 0
            FSControl1.setValue fsGear, 1
            FSControl1.setValue fsThrottle1, N2R
            FSControl1.setValue fsThrottle2, N2L
        End If
```

4. Upon arrival to 120 meters altitude, begins preparing for the flying phase by run the autopilot and provide it with the required parameters values so as to control the speed of the aircraft.

```
If Altitude >= (FSAirportAlt + 120) Then
        FSControl1.setValue fsElevator, 0
        FSControl1.setValue fsFlaps, 0
        FSControl1.setValue fsGear, 1
        FSControl1.setValue fsAP, 1
        FSControl1.setValue fsAThrottle, 1
        FSControl1.setValue fsASpeedH, 1
          If Heading >= 180 Then
             Heading = -(360 - Heading)
          End If
        FSControl1.setValue fsHeading2, Heading
        AltDiff = FSAirportAlt - FDRAirportAlt
        FSControl1.setValue fsAlt, (Altitude + AltDiff)
        SpeedBfr = 0
```

5. Declare the end of the take-off phase and the beginning of the flying phase, therefore terminate Timer2 and activates Timer3 with time interval 100ms.

```
        Timer2.Interval = 0
        Timer3.Interval = 100
```

3.5.2.4.2. Flying

Flying stage simulate the real flight mission with high accuracy, the flight parameters values are sent to the flight simulator each 100ms. This stage goes through the following steps within each cycle of Timer3 after Timer2 is terminated and activates Timer3.

1. Check the end of data file and the flying stage continuality with the check of position of Landing gears

```
        If Not EOF(iFileNoR1) And LG = 1 Then
```

2. Read one line from the parameter's data file, then split the text line in a string array and extract the parameters values.

```
Input #iFileNoR1, TextFileNo1
strSpilit() = Split(TextFileNo1, "|")
F_Time = strSpilit(0)
Speed = strSpilit(1)
Altitude = strSpilit(2)
Heading = strSpilit(3)
   If Heading >= 180 Then
      Heading = -(360 - Heading)
   End If
Roll = strSpilit(4)
Pitch = strSpilit(5)
LG = strSpilit(11)
```

3. Send the values of the flight parameters to the flight simulator program.

```
FSControl1.setValue fsAlt, (Altitude + AltDiff)
FSControl1.setValue fsHeading2, Heading
FSControl1.setValue fsBank2, Roll
FSControl1.setValue fsPitch2, Pitch
```

4. If the aircraft speed changed, then the speed value is sent to the autopilot to maintain the aircraft speed with the new value by controlling the Throttle Levers.

```
Private Sub Speed_Change()
      On Error Resume Next
      FSControl1.setValue fsASpeed, Speed
End Sub
```

5. If the Landing gears of the simulated data are extended, the flying stage is ended, thus extend the flight simulator's aircraft Landing gears and Flaps and terminate Timer3 and activate Timer4 with time interval 100ms.

```
Else
      FSControl1.setValue fsFlaps, 1
      FSControl1.setValue fsGear, 0
      Timer3.Interval = 0
```

```
                    Timer4.Interval = 10
                        x = 100
                z:   End If
```

3.5.2.4.3. Landing

The landing phase is separated from the flying phase to reduce the number of parameters that is being sending to the simulation program every 100 ms. This is because the difference between the two phases is the position of Landing gear and Flaps (retracted during flying and extended during landing), so the change to the Landing gear and Flaps has been placed before the end of the flying stage in the preceding paragraph (**3.5.2.4.2** - 5). Landing stage goes through the following steps within each cycle of Timer4 after Timer3 is terminated and activates Timer4.

1. Check the end of data file and the landing stage continuality with check of the simulated data altitude comparing to the flight simulator airport altitude

```
        If Not EOF(iFileNoR1) And (Altitude + AltDiff) <> FSAirportAlt Then
```

2. Read one line from the parameter's data file, then split the text line in a string array and extract the parameters values.

```
        Input #iFileNoR1, TextFileNo1
        strSpilit() = Split(TextFileNo1, "|")
        F_Time = strSpilit(0)
        Speed = strSpilit(1)
        Altitude = strSpilit(2)
        Heading = strSpilit(3)
           If Heading >= 180 Then
               Heading = -(360 - Heading)
            End If
        Roll = strSpilit(4)
        Pitch = strSpilit(5)
```

3. Send the values of the flight parameters to the flight simulator program.

```
        FSControl1.setValue fsAlt, (Altitude + AltDiff)
```

FSControl1.setValue fsHeading2, Heading

FSControl1.setValue fsBank2, Roll

FSControl1.setValue fsPitch2, Pitch

4. If the simulated data altitude equal to the flight simulator airport altitude, the flying stage is ended, thus after 10s pause the MS FS program and terminate to stop simulation.

```
    Else

            x = x - 1
            If x = 0 Then

                    FSControl1.setValue fsPause, 1
                    Timer4.Interval = 0
                    MsgBox ("Open Flight Analysis" &vbCrLf &_
                      "Go to menu Options\Flight Analysis")

            End If

    z:
    End If
```

Chapter Four
Results and Discussions

4.1. Introduction

Although, the Тестер УЗ-ЛК flight data recorder is the recorder of many Soviet Union aircraft types operated in Sudan, the use of Dvina-M software produced with this type of recorder for data analysis is fraught with major limitations and inabilities. These deficiencies of the analysis software can be justified with the orientation of aircraft manufacturers to monopolize the economic benefits of pilots training, lack the software ability to analyse pilot's training flights lead to higher costs of training in countries operating the aircraft, therefore, these countries forced to send their pilots to the aircraft manufacturers for training.

From this perspective, the most important result was expected from this research is to build a system can analyse pilot's flight training sessions, to reduce the effort, time and cost, with the ability to raise the capabilities and skills of pilot's competencies.

4.2. Results

The Real Flight Simulation program produces an interactive environment, enables conversion of raw flight data to movement appears on an aircraft model, which interprets the data extracted from the FDR. The Real Flight Simulation program focused on the instructors and pilots requirements to analyse the flights sessions which are:

1. Aircraft model simulates the aircraft motion and maneuvers.
2. Measures of cockpit instruments during flight.
3. Flight area maps with navigational charts.
4. Flight path and flights zones.

Therefore, the Real Flight Simulation used to possess the capabilities to control the Microsoft Flight Simulator program in order to extract the pilot training features underlying of the Microsoft Flight Simulator as a pilot's

training aid. The Real Flight Simulation program able to translate the flight data through the following MS FS characteristics:

1. The ability of start simulation from the airport that have been used in the real flight (Khartoum, Wadi-saydna, Nyala, Portsudan, etc), with accurate takeoff heading and real flying zones. Figure (4.1) shows the Khartoum Airport as example.

Figure (4.1): *Khartoum Airport*

2. Provide deferent types of view during simulation process (Spot view, Tower view, Cockpit view, Virtual-cockpit view and Top-Down view), with the ability to manage the characteristics of these views to increase its usability, like look around direction, zoom in/out, eye-point position, ect. The figures (4.2-a) to (4.2-e) show the different views of simulation and the ability to manage their characteristics.

Figure (4.2-a): *Spot view: provides the aircraft view from the outside, as it would from another aircraft. Reposition the spot plane, gives the ability to analyse the flight from any angle of view.*
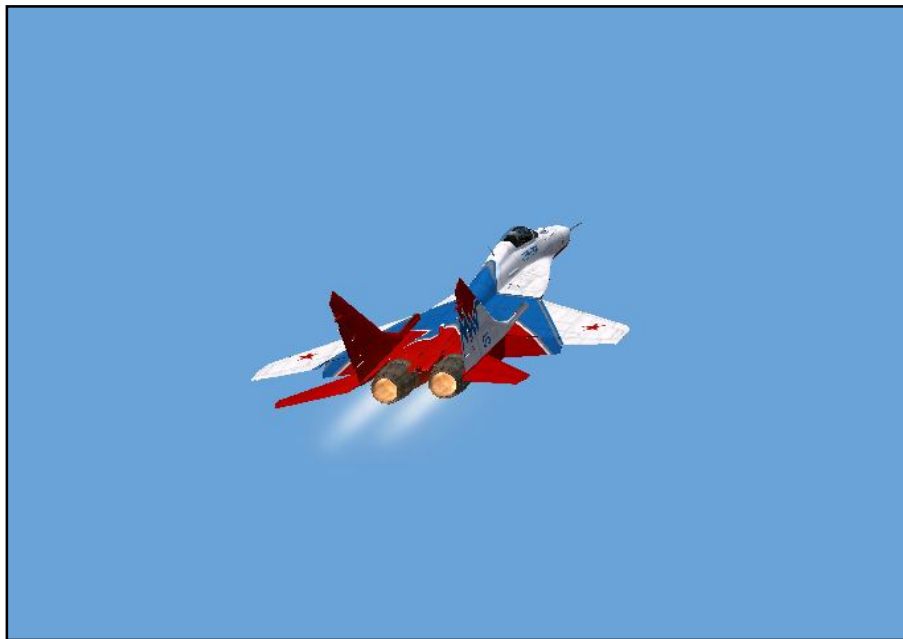

Figure (4.2-b): *Tower view: provides the view of the aircraft as it looks from the closest air traffic control tower.*

Figure (4.2-c): *Cockpit view: enables both the 3-D world outside windscreen and the detailed 2D instrument panels or only the primary flight instruments.*

Figure (4.2-c): *Virtual-cockpit view: puts the pilot's eyes inside a three-dimensional aircraft's cockpit with ability to look around.*

Figure (4.2-d): *Top-Down view: puts the pilot's eyes literally, above it all, with property to modify the map orientation and altitude.*
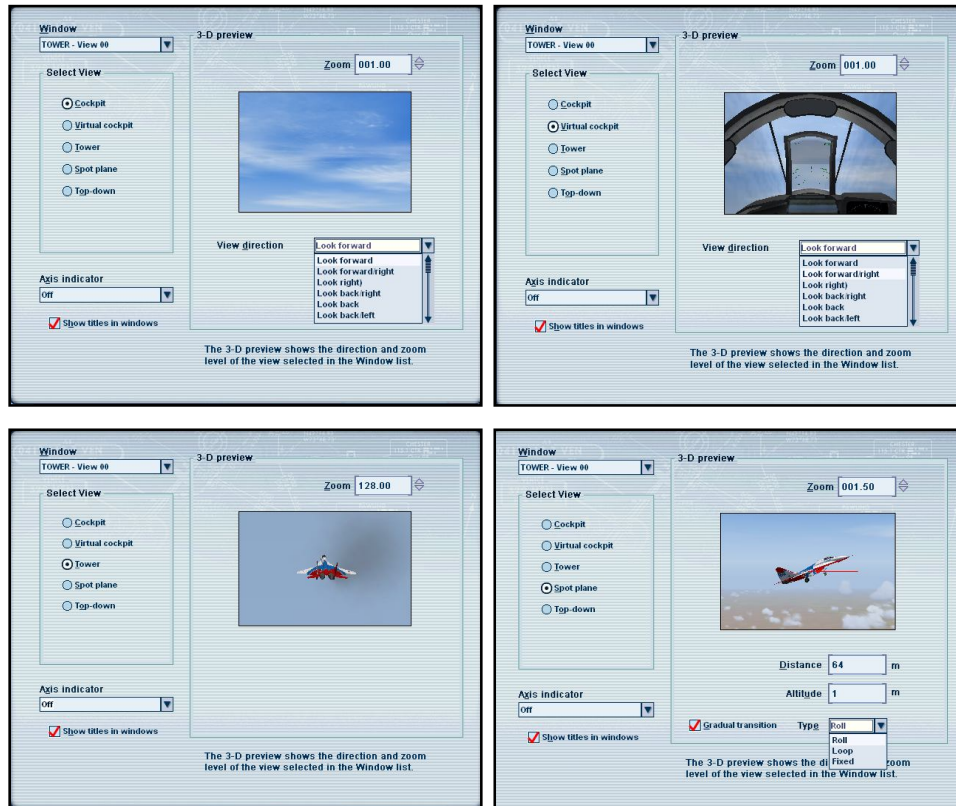

Figure (4.2-e): *Managing the characteristics of the views to increase its usability*

3. Global Positioning System (GPS), used to increase the efficiency of flight. With its accurate, continuous, and global capabilities, GPS offers seamless satellite navigation services that satisfy the pilot's requirements for navigation. Space-based position and navigation enables three-dimensional position determination for all phases of flight from departure, en route, and arrival, to airport surface navigation.

Figure (4.3): *Global Positioning System (GPS)*

The *Garmin GPS 500* unit in Flight Simulator can help pilots for:

➢ Continual determination of aircraft position over the map.

➢ Determine where the destination is.

➢ Determine how to get from current location to destination.

➢ View of a graphic depiction of the terrain.

➢ Provides information and locations of airports, intersections, NDBs, and VORs.

➢ Follow a VFR or IFR flight plan.

> ➤ Fly instrument procedures.

> ➤ Be aware of the airspace boundaries in vicinity.

4. Simulation flight recorder, which records the simulation session to the MS FS main folder, making the flight simulation sessions available for future analysis and discussions and movable to the computers have the Microsoft Flight Simulator. Figure (4.4) shows the Flight Recorder.
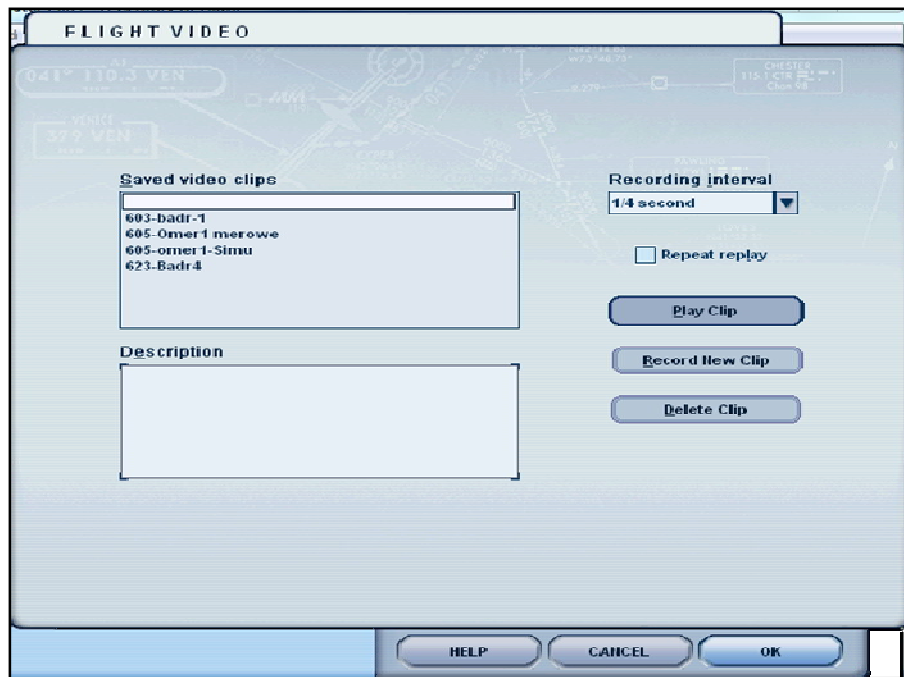

Figure (4.4): *The Flight Recorder.*

5. At the end of flight simulation, or before the end, the Flight Analysis option provides the flight path on the map view with real coordination (real Latitudes and Longitudes), altitude and airspeed. The flight path can be replayed, forwarded and backwarded to redraw the simulation session path. Figures (4.5-a) & (4.5-b) show paths of flight sessions.
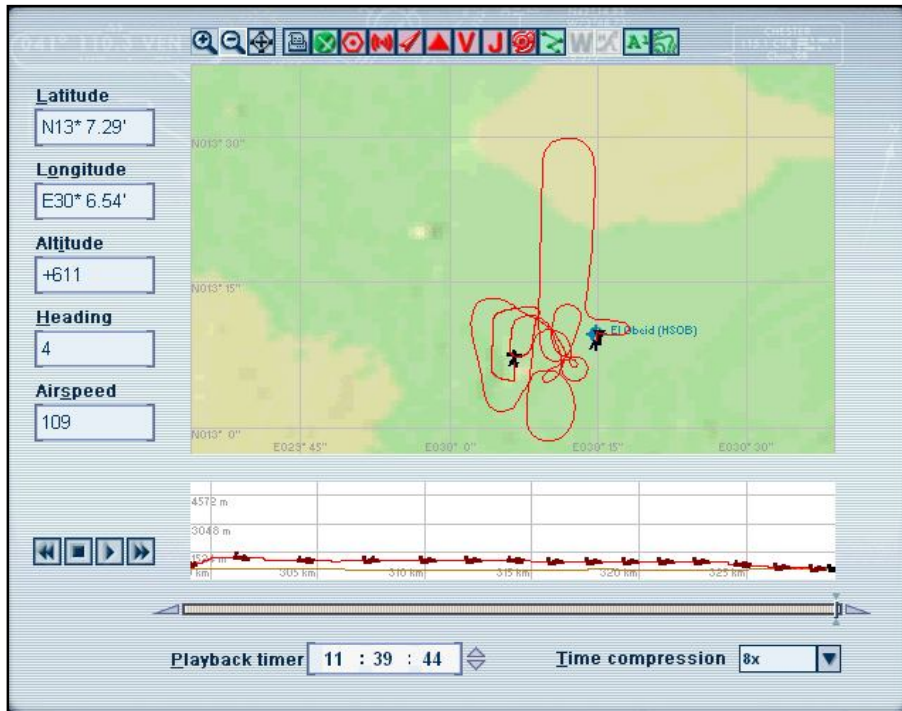
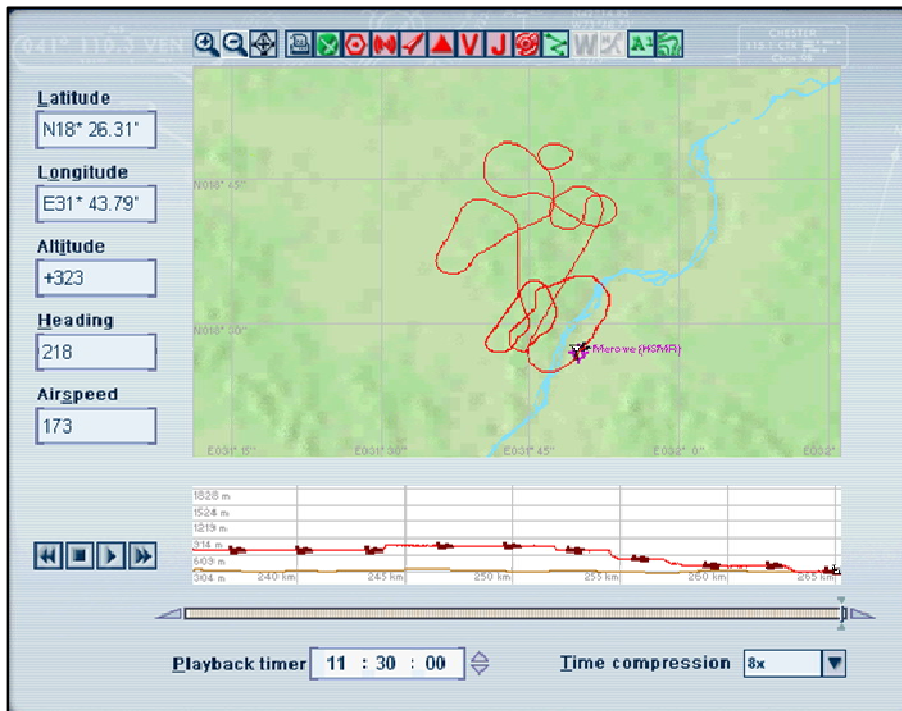Figure (4.5-a): *Before simulation end flight path, of a flight session started from El Obeid airport.*

Figure (4.5-b): *After simulation end flight path, of a flight session started from Merowe airport*

The previous characteristics of the Ms-FS forms an interactive flight analysis environment for instructors and pilots, as a result of full control of the Real Flight Simulation program over the Ms-FS, with high accuracy data, and the ability to send and translate the flight parameter's values to movement of the aircraft model during period amounted to 0.1 second, even in cases of high maneuverability.

4.3. Discussion

The Real Flight Simulation software is an instructor pilot tool to visualize all the flight training sessions, so that instructors and pilots can monitor, analyse, asses, discus and preplan the future training flights. With the use of Real Flight Simulation software, instructors and pilots never need to look at tables and charts of hundreds flight parameters and asses thousands/millions of flight's parameters values recorded by the flight data recorder. The software provides full combinations of views, which create embodiment, more than what is going on the imagination of instructors and pilots, whereas, the software enables the instructors and pilots to open a large number of windows options, that can be seen in a single moment, allowing the possibility of monitoring and evaluating the performance from different aspects. Figures (4.6-a) to (4.6-b) show how the combinations of windows can be formed.
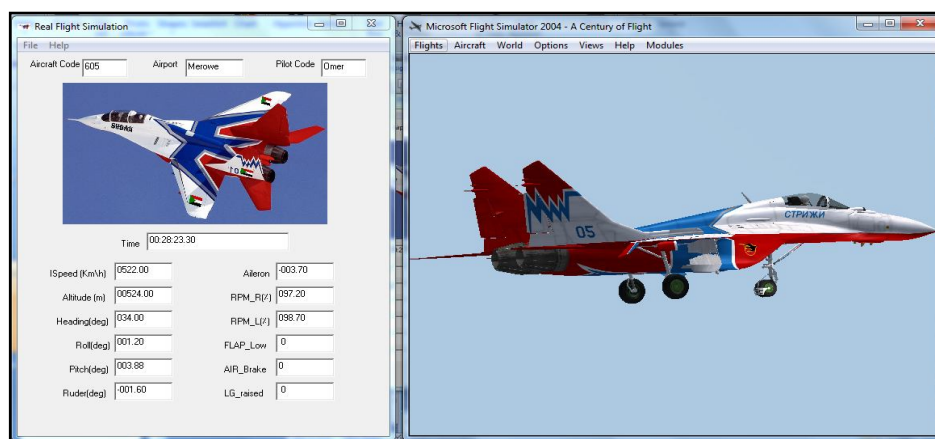

Figure (4.6-a): The *Real Flight Simulation's window combined with the Tower View*

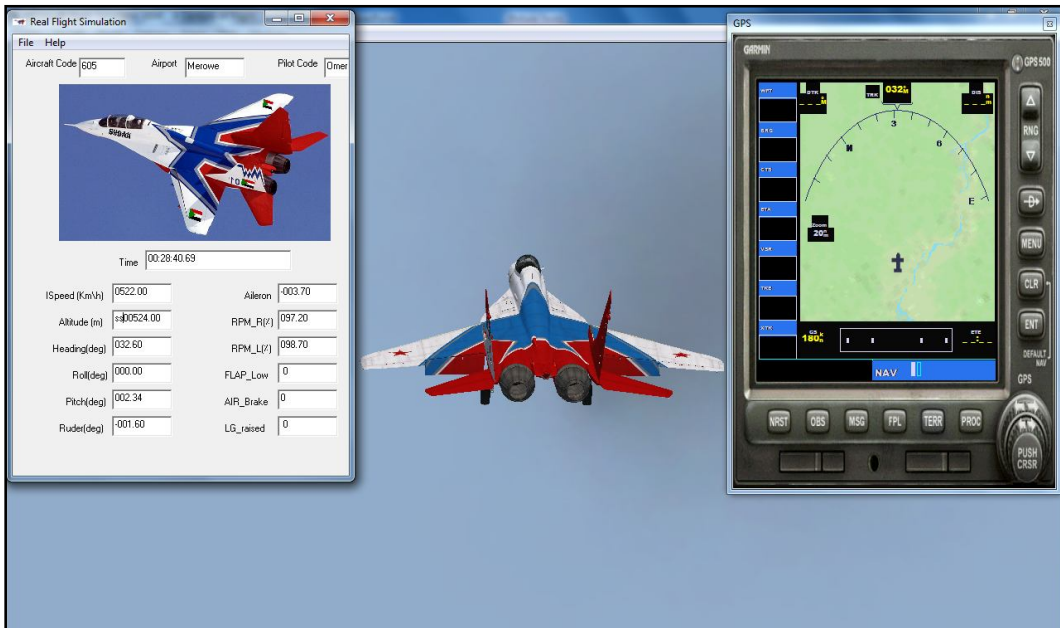Figure (4.6-b): *Combination of the Virtual-Cockpit, Spot and GPS with the main window Cockpit View*

Figure (4.6-c): *Combination of the Real Flight Simulation and GPS with the main window Spot View*

Figure (4.6-c): *Combination of the Spot, Top-Down, Real Flight Simulation and GPS with the main window Cockpit View-Basic instruments*

The program eliminates the weaknesses associated with the Тестер У3-ЛК flight data recorder, which hindered the work of flying crews that their aircrafts operate recorder of this type. As mentioned earlier, few parameters which are recorded on the Тестер У3-ЛК flight data recorder are suitable for analysis of the flight training sessions, this makes taking the advantage of these parameters for flight analysis is almost impossible. Hence, the power of the Real Flight Simulation software is the conversion of a fraction volume of the parameters data stored in a small text file, to a tremendous amount of ways and means for the analysis of flight training sessions.

The Software reduce the instructor's effort of teaching, when using the boards and markers to visualize the flight session, also, eliminates the student's pilots pain of imagination, so as to understand what the instructors trying to visualize. This is reflected on the overall duration of the aircraft Conversion Courses, which reduces the needed time for a pilot to complete his study, with a significant increase in the student's pilots

understanding level for the flying lessons, which raise their capabilities and skills during practice on the aircraft by observing, watching and connect cause and effect relationships when the instructor uses the Real Flight Simulation on the "After flight briefing".

The great impact of using the Real Flight Simulation is evident in training cost, reduce the training period is reducing the number of flights required by the trainee pilot to reach the level of individual leadership of the aircraft with advanced skills and high maneuvers capabilities. The cost of training flight is between 7,500$ and 11,000$ for the aircrafts that use the Тестер У3-ЛК flight data recorder. Therefore, reduce the number of flights have a significant impact on the total cost of flying crews training.

Chapter Five

Conclusions and Recommendations

5.1. Conclusions

The main goal of the current project was to develop software that rearranges the flights parameter's values recorded in Тестер УЗ-ЛК flight data recorder so as to simulate the flight session and reproduces the flight trajectory. This software will be used by instructors and pilots as a tool for Flight Data Analysis and Management process to monitor, evaluate and correct the pilot's behavior during the flight training sessions.

The Real Fight Simulation was designed to accomplish this task by interprets the data extracted from the FDR and reform the parameters of the flight data recorder to meet the specifications of the Microsoft Flight Simulator parameters, in order to pass the flight parameters through the FSUIPC to control the Microsoft Flight Simulator so as to simulate the aircraft behavior and motion during the real flight sessions.

5.2. Recommendations

Flight Data Analysis or Management of Flight Data is one of the fields that combine Aeronautics and Computer Science, especially after the great development occurs in aircraft systems and navigational devices and their dependency on computerized systems. This research is an effort in the same context, so as to provide a great benefit and great service in the field of pilots' training to provide efficient pilots with high skills, although reduced training costs by as much as 60%.

This research can replace the purchase orders of flights analysis softwares with high prices from aircraft manufacturers, which affects positively on the aircraft's operators. Also those operators can use the Real Flight Simulation program in the investigation of aircraft accidents, to identify their causes and their relation to the mistakes of the pilots.

This research is the beginning of a research project to develop an integrated design for an aircraft simulator, which represents one of the most important liabilities for the pilots training, therefore to raise the efficiency, reduce the training cost and Increases the life span of the aircrafts.

REFERENCES

Daniel P. Raymer, 2002, Aircraft Design : Conceptual Approach, 2nd edition AIAA, California-USA.

David F. Anderson and Scott Eberhardt, 2001, Understanding Flight, 1st edition McGraw-Hill, USA.

EHJ Pallett, 1988, Aircraft Instruments, 2nd edition, UK.

Francis J. Hale, 1984, Introduction to aircraft performance, selection and design, 1st edition John Wiley & Sons, USA.

Jan Roskam, 2001, Airplane Flight Dynamics and Automatic Flight Controls, 3rd edition DARcorporation, USA.

MiG Corporation, 2001, Maintenance Manual, Russia.

Pete Dowson, 2012, FSUIPC for Programmers, 4th edition, USA.

Microsoft Corporation (2014), MSDN Library, Available at: http://msdn.microsoft.com/en-us/library, Accessed 7 June 2014.

Peter Dowson (2013), FSUIPC_SDK, Available at: http://www.schiratti.com/dowson.html, Accessed 29 April 2014.