# GPS BASED REMOTE BLACK BOX FOR AIRBORNE VEHICLES

*** Adarsh Kumar H K, Dr. Ramachandra C G**
**** Dr. Srinivas T R,**
***** Rishi J P,**
****** Virupaxappa B**
** Department of Mechanical Engineering*
*Srinivas Institute of Technology*
*Mangaluru, India*
*** Department of Industrial and Production Engineering*
*Sri Jayachamarajendra College of Engineering*
*Mysuru, India*
**** Department of Mechanical Engineering*
*Vidyavardhak College of Engineerng*
*Mysuru, India*
***** Department of Mechanical Engineering*
*Dr M.V. Shetty Institute of Technology*
*Moodbidri, India.*

## ABSTRACT

*Modern day Flight Data Recorders receive inputs via specific data frames from the flight-data acquisition units. They record significant flight parameters, including the control and actuator positions, engine information and time of day. There are 88 parameters required as a minimum under current U.S. Federal regulations. But some systems monitor many more variables. Generally each parameter is recorded a few times per second, though some units store "bursts" of data at a much higher frequency if the data begins to change quickly. Most FDRs record approximately 17–25 hours worth of data in a continuous loop. It is proposed to design minimal flight recorder with 5 to 6 parameters like temperature of the engine, engine noise, vibration, position, altitude etc. And transmit the data to the remote server. This data can be sent by wireless technology every 10 seconds and can be modified later on with advanced wireless communication. It is proposed to use GPS device to find out the aircraft position and altitude of the aircraft. Using advanced RISC microcontroller and electronic sensor other parameters mentioned are monitored. Using wireless modem all the data is sent to base station, another PC server with wireless modem. Here data is continuously received every predefined timing and logged into database for future analysis.*

*Keywords: GPS; Black Box; Airborne Vehicle; Flight Data; Flight Parameters*:

## INTRODUCTION

Popularly referred to as a "black box", the data recorded by the FDR [01] is used for accident investigation, as well as for analyzing air safety issues, material degradation and engine performance. Due to their importance in investigating accidents, these International Civil Aviation

Organization (ICAO)-regulated devices are carefully engineered and stoutly constructed to withstand the force of a high speed impact and the heat of an intense fire. Contrary to the "black box" reference, the exterior of the FDR is coated with heat-resistant bright Red paint for high visibility in wreckage, and the unit is usually mounted in the aircraft's empennage (tail section), where it is more likely to survive a severe crash. Following an accident, recovery of the "black boxes" is second in importance only to the rescue of survivors and recovery of human remains. This has also given rise to flight data monitoring programs, whereby flights are analyzed for optimum fuel consumption and dangerous flight crew habits. The data from the FDR is transferred, in situ, to a solid state recording device and then periodically analyzed with some of the same technology used for accident investigations. FDRs are usually located in the rear of the aircraft, typically in the tail. In this position, the entire front of the aircraft is expected to act as a "crush zone" to reduce the shock that reaches the recorder. Also, modern FDRs are typically double wrapped, in strong corrosion-resistant stainless steel or titanium, with high-temperature insulation inside. They are usually bright orange. They are designed to emit a locator beacon for up to 30 days, and can operate immersed to a depth of up to 6,000 meters (20,000 ft).

A flight data recorder (FDR) (also ADR, for accident data recorder) is a kind of flight recorder [02]. It is a device used to record specific aircraft performance parameters. Another kind of flight recorder is the cockpit voice recorder (CVR), which records conversation in the cockpit, radio communications between the cockpit crew and others (including conversation with air traffic control personnel), as well as ambient sounds. In some cases, both functions have been combined into a single unit. The current applicable FAA TSO is C124b titled Flight Data Recorder Systems. Popularly referred to as a "black box", the data recorded by the FDR is used for accident investigation, as well as for analyzing air safety issues, material degradation and engine performance. Due to their importance in investigating accidents, these International Civil Aviation Organization (ICAO) regulated devices are carefully engineered and stoutly constructed to withstand the force of a high speed impact and the heat of an intense fire. Contrary to the "black box" reference, the exterior of the FDR is coated with heat-resistant bright Red paint for high visibility in wreckage, and the unit is usually mounted in the aircraft's empennage (tail section), where it is more likely to survive a severe crash. Following an accident, recovery of the "black boxes" is second in importance only to the rescue of survivors and recovery of human remains.

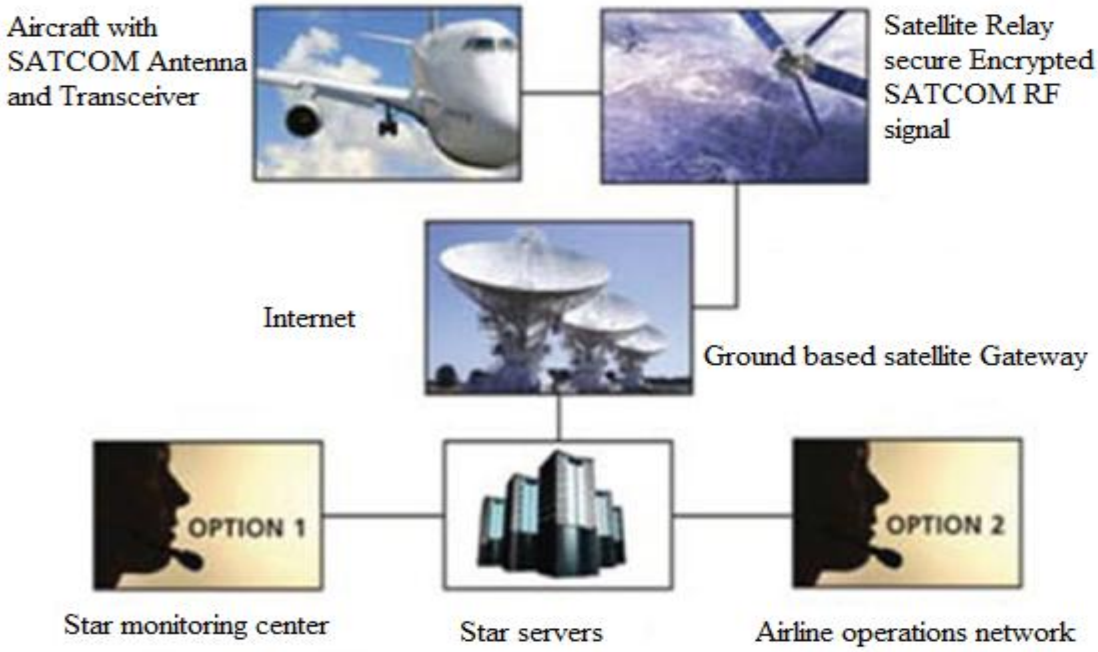



Figure 1. Terra star transmits collected data during the flight.

Experimental results demonstrate that structure [03] computation as a tool for black-box modeling may be a useful tool for the analysis of dynamic aircraft data. The LASSO (least absolute shrinkage and selection operator) successfully reduced the number of repressors posed to aircraft aero elastic data yielding a parsimonious model structure for each data set. Additionally, these parsimonious structures were capable of predicting a large portion of the cross-validation data, collected on the adjacent wing and with a different sensor. This suggests that the identified structures and parameters explain the data well. Using percent fit alone as an indicator of model goodness could lead to incorrect interpretations of model validity. However, in many cases, for nonlinear models this may be the only indicator that is readily available.

## ANALYSIS AND METHODOLOGY

This project works based on information or data will continuously storing in remote pc server and its keep on update the information every 10 seconds in base station. There is various devices are using in our project, are as shown in block diagram.

## BLOCK DIAGRAM

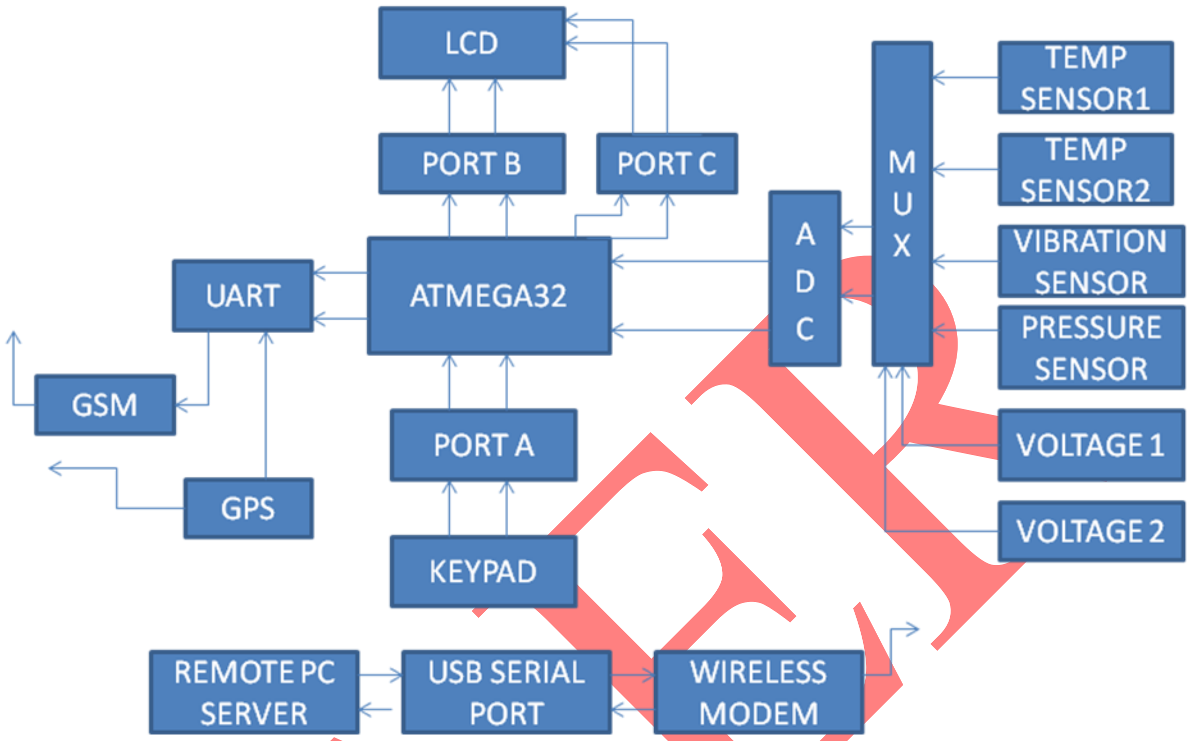



Figure 2. Block diagram of remote black box.

The ATmega32 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega32 achieves throughputs approaching 1 MIPS per MHz allowing the system designed to optimize power consumption versus processing speed. High-performance, Low-power AVR 8-bit Microcontroller Advanced RISC Architecture, 131 Powerful Instructions – Most Single-clock Cycle Execution , 32 x 8 General Purpose Working Registers. Up to 16 MIPS Throughput at 16 MHz 32K Bytes of In-System Self-Programmable Flash Endurance: 10,000 Write/Erase Cycles, Optional Boot Code Section with Independent Lock Bits In-System Programming by On-chip Boot Program,1024 Bytes EEPROM Endurance: 100,000 Write/Erase Cycles, 2K Byte Internal SRAM, Programming Lock for Software Security, Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface, Peripheral Features– Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes– One 16-bit Timer/Counter with Separate Prescaler, 8-channel, 10-bit ADC Etc..

### A. *Universal Asynchronous Receiver/Transmitter*

A Universal Asynchronous Receiver/Transmitter (UART) is a piece of computer hardware [08] that translates data between parallel and serial forms. UARTs are commonly used in conjunction with communication standards such as EIA, RS-232, RS-422 or RS-485.The universal designation indicates that the data format and transmission speeds are configurable and that the actual electric signaling levels and methods typically are handled by a special driver circuit external to the UART. A UART is usually an individual (or part of an) integrated circuit used for serial communications over a computer or peripheral device serial port. UARTs are now commonly included in microcontrollers. A dual UART, or DUART, combines two UARTs into a single chip. Many modern ICs now come with a UART that can also communicate synchronously; these devices are called USARTs (universal synchronous/asynchronous receiver/transmitter).

### *C. Analog-To-Digital Converter*

An analog-to-digital converter (ADC) is a device that converts a continuous quantity to a discrete time digital representation. An ADC may also provide an isolated measurement. The reverse operation is performed by a digital-to-analog converter (DAC).Typically, an ADC is an electronic device that converts an input analog voltage or current to a digital number proportional to the magnitude of the voltage or current. However, some non-electronic or only partially electronic devices, such as rotary encoders, can also be considered ADCs. The digital output may use different coding schemes. Typically the digital output will be a two's complement binary number that is proportional to the input, but there are other possibilities. An encoder, for example, might output a Gray code.

### D. *Sensors*

Sensors are devices used to provide information on the presence or absence of an object [04]. Measurement of temperature is critical in modern electronic devices, especially expensive laptop computers and other portable devices with densely packed circuits which dissipate considerable power in the form of heat. Knowledge of system temperature can also be used to control battery charging as well as prevent damage to expensive microprocessors. Compact high power portable equipment often has fan cooling to maintain junction temperatures at proper levels. In order to conserve battery life, the fan should only operate when necessary. Accurate control of the fan requires a knowledge of critical temperatures from the appropriate temperature sensor. Accurate temperature measurements are required in many other measurement systems such as process control and instrumentation applications. In most cases, because of low level nonlinear outputs, the sensor output must be properly conditioned and amplified before further processing can occur. Except for IC sensors, all temperature sensors have nonlinear transfer functions. In the past, complex analog conditioning circuits were designed to correct for the sensor nonlinearity. These circuits often required manual calibration and precision resistors to achieve the desired accuracy. Today, however, sensor outputs may be digitized directly by high resolution analog-to-digital converters (ADCs). Linearization and calibration is then performed digitally, thereby reducing cost and complexity.

## SYSTEM REQUIREMENT SPECIFICATIONS

### *A. Lab View*

This project is aimed to store the data in personal computer. We need to use technology related with intensive processing and manipulations. Lab VIEW (Laboratory Virtual Instrument

Engineering Workbench) is a development environment based on the graphical programming language *G*. LabVIEW is integrated fully for communication with hardware such as GPIB, VXI, PXI, RS-232, RS-485, and Plug-in data acquisition boards. LabVIEW also has built-in libraries for using software standards such as TCP/IP Networking and ActiveX. Using LabVIEW, we can create 32-bit compiled programs that give us the fast execution speeds needed for custom data acquisition, test, and measurement solutions. There are hundreds of tools and built in functions for graph, manipulations and processing functions in Lab VIEW. Using these functions it is possible to implement and write effective program to achieve our goal of successful to receive the data to base station and search system. Lab VIEW is a programming language and platform, which is highly suitable for engineering, scientific and R & D applications. It will helps view how the information transfer from source to destination. To achieve this goal Lab VIEW platform is selected. The Lab VIEW will display the data flow graph and bit rates of object successful.

### B. *Virtual Instruments*

Virtual instruments (VI) are the building blocks of Lab VIEW programming. Lab VIEW programs are called as VIs because they have the look and feel of physical systems or instruments. A VI and its components are analogues to main programs and sub routines from text programming languages like C and FORTAN. VIs have both an interactive user interface- known as the front panel and the source code represented in graphical form on the block diagram. Lab VIEW provides mechanisms that allow data to pass easily between the front panel and the block diagram. The art of successful programming in G is an exercise in modular programming. After dividing a given task into a series of simpler subtasks (in G these subtasks are called sub VIs) we can then construct a virtual instrument to accomplish each subtask. The resulting subtasks are then assembled on a top-level block diagram to form the complete program. Modularity means that you can execute each sub VI independently, thus making debugging and verification easier. Furthermore, if sub VIs are general-purpose programs, they can be used in other programs.

VIs (and sub VIs) has three main parts: the front panel, the block diagram, and the icon/connector. The front panel is indispensable for viewing the program outputs. The front panel contains knobs, push buttons, graphs and many other controls and indicators. The block diagram is the source code for the VI. The source code is “written” in the G programming language. The block diagram is actually the executable code. The icons of a block diagram represent lower level VIs, built in functions, and program control structures. These icons are wired together to allow the data flow. This concept is called as data flow programming. The icons and connectors specify the pathways for the data to flow into and out of the VIs. The icon is the graphical representation of the VI in the block diagram and the connector defines the inputs and the outputs. All VIs have an icon and a connector.

## SYSTEM HARDWARE

**1. AT Mega 32:** High-performance, Low-power AVR 8-bit Microcontroller. The pin configuration of AT Mega 32 is as follows

**VCC:** Digital supply voltage

**GND**: Ground.

**Port A (PA7….PA0):** Port A serves as the analog inputs to the A/D Converter. Port A also serves as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

**Port B (PB7….PB0):** Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running. Port B also serves the functions of various special features of the ATmega32 as listed.

**Port C (PC7….PC0):** Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PC5 (TDI), PC3 (TMS) and PC2 (TCK) will be activated even if a reset occurs. The TD0 pin is tri-stated unless TAP states that shift out data are entered. Port C also serves the functions of the JTAG interface and other special features of the ATmega32 as listed.

**Port D (PD7….PD0):** Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit).The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running. Port D also serves the functions of various special features of the ATmega32 as listed.

**RESET**: Reset Input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a reset.

**XTAL1:** Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

**XTAL2:** Output from the inverting Oscillator amplifier.

**AVCC:** AVCC is the supply voltage pin for Port A and the A/D Converter. It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter.

**AREF:** AREF is the analog reference pin for the A/D Converter.

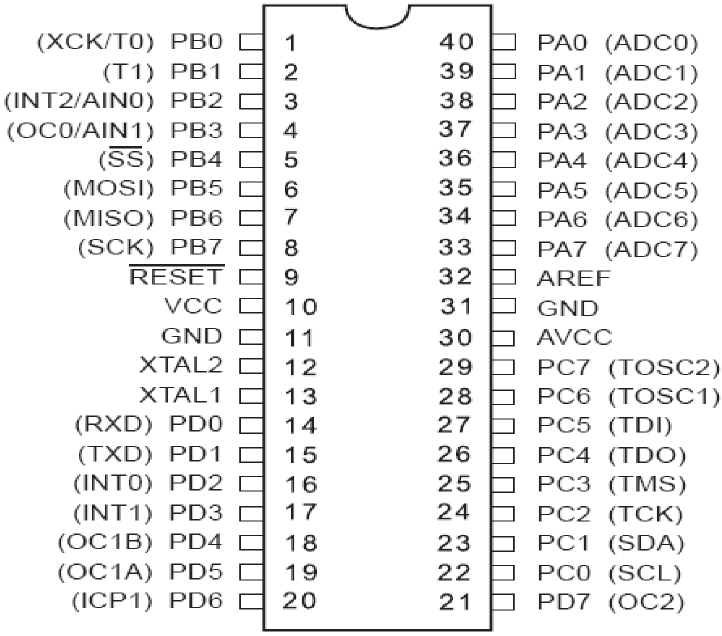



Figure 3. Pin Diagram AT Mega 32.

**2. AVR MCU Architecture:** In order to maximize performance and parallelism [07], the AVR uses Harvard architecture– with separate memories and buses for program and data. Instructions in the program memory are executed with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is In- System Reprogrammable Flash memory.

**3. ALU – Arithmetic Logic Unit:** The high-performance AVR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations between general purpose registers or between a register and an immediate are executed. The ALU operations are divided into three main categories – arithmetic, logical, and bit-functions. Some implementations of the architecture also provide a powerful multiplier supporting both signed/unsigned multiplication and fractional format. "Instruction Set" section provides detailed description.

**4. EEPROM Data Memory:** The ATmega32 contains 1024 bytes of data EEPROM memory. It is organized as a separate data space, in which single bytes can be read and written. The EEPROM has endurance of at least 100,000 write/erase cycles. The access between the EEPROM and the CPU is described in the following, specifying the EEPROM Address Registers, the EEPROM Data Register, and the EEPROM Control Register.

**5. External Interrupts:** The External Interrupts are triggered by the INT0, INT1, and INT2 pins. Observe that, if enabled, the interrupts will trigger even if the INT0….2 pins are configured as outputs. This feature provides a way of generating a software interrupt. The external interrupts can be triggered by a falling or rising edge or a low level (INT2 is only an edge triggered interrupt).

This is set up as indicated in the specification for the MCU Control Register – MCUCR – and MCU Control and Status Register – MCUCSR. When the external interrupt is enabled and is configured as level triggered (only INT0/INT1), the interrupt will trigger as long as the pin is held low.

### 6. Analog to Digital Converter

### 7. LCD Controllers (LM018L)

A liquid crystal display is a low cost device capable of displaying text and images. LCD are extremely common in embedded system since such system often do not have video monitor like those that come standard with desktop system. LCD can be found in numerous common devices like watches, fax and copy machines and calculators.

### EMBEDDED SYSTEM

An embedded system is a combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a specific function By definition all embedded systems contain a processor and software, but what other features do they have in common? Certainly, in order to have software, there must be a place to store the executable code and temporary storage for runtime data manipulation. These take the form of ROM and RAM, respectively; any embedded system will have some of each. If only a small amount of memory is required, it might be contained within the same chip as the processor. Otherwise, one or both types of memory will reside in external memory chips.

All embedded systems also contain some type of inputs and outputs [09]. For example, in a microwave oven the inputs are the buttons on the front panel and a temperature probe, and the outputs are the human-readable display and the microwave radiation. It is almost always the case that the outputs of the embedded system are a function of its inputs and several other factors (elapsed time, current temperature, etc.). The inputs to the system usually take the form of sensors and probes, communication signals, or control knobs and buttons. The outputs are typically displays, communication signals, or changes to the physical world.

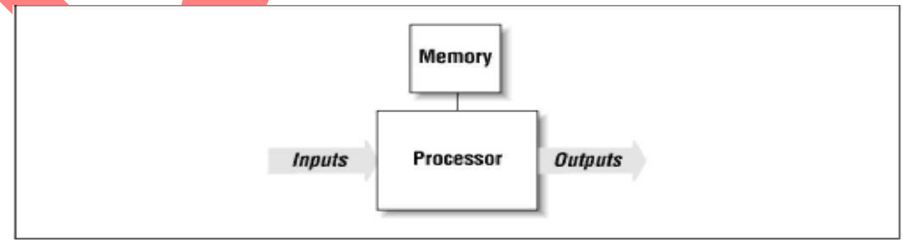



Figure 4. A Generic Embedded System.

## SYSTEM SOFTWARE

### A. *AVR Studio*

The AVR Lib c package provides a subset of the standard C library for Atmel AVR 8-bit RISC microcontrollers. In addition, the library provides the basic startup code needed by most applications. There might be several reasons to write code for AVR microcontrollers using plain assembler source code. Among them are:

- Code for devices that do not have RAM and are thus not supported by the C compiler.
- Code for very time-critical applications.
- Special tweaks that cannot be done in C. Usually, all but the first could probably be done easily using the inline assembler facility of the compiler. Although avr-libc is primarily targeted to support programming AVR microcontrollers using the C (and C++) language, there's limited support for direct assembler usage as well. The benefits of it are:
- Use of the C preprocessor and thus the ability to use the same symbolic constants that are available to C programs, as well as a flexible macro concept that can use any valid C identifier as a macro (whereas the assembler's macro concept is basically targeted to use a macro in place of an assembler instruction).
- Use of the runtime framework like automatically assigning interrupt vectors. For devices that have RAM, initializing the RAM variables can also be utilized.

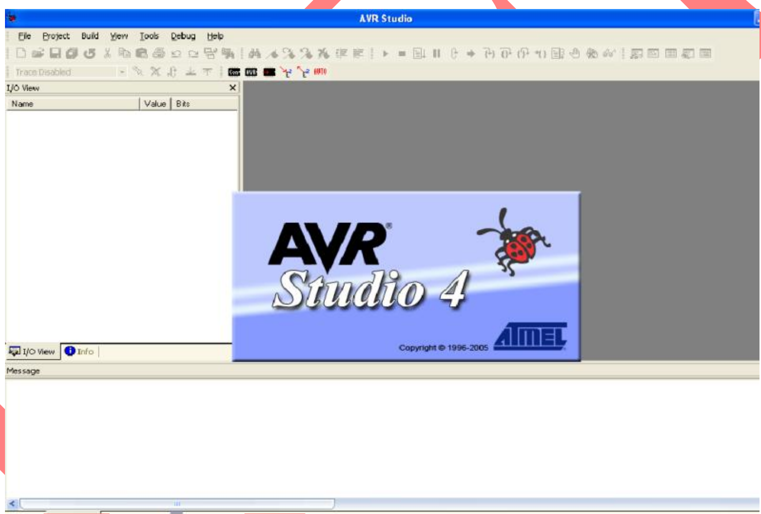



Figure 5. AVR Studio 4.

## NETWORK COMMUNICATION TECHNOLOGY

### A. *Global Positioning System*

Global Positioning System is a highly integrated smart GPS [06] module with a ceramic GPS patch antenna. The antenna is connected to the module. The module is with 51 channel acquisition engine and 14 channel track engine, which be capable of receiving signals from up to 65 GPS satellites and transferring them into the precise position and timing information that can be read over either UART port or RS232 serial port. Small size and high-end GPS functionality are at low power consumption, Both of the LVTTL-level and RS232 signal interface are provided on the interface connector, supply voltage of 3.6V~6.0V is supported. The smart GPS antenna module is available as an off-the-shelf component, 100% tested. The smart GPS antenna module can be offered for OEM applications with the versatile adaptation in form and connection. Additionally, the antenna can be tuned to the final systems' circumstances.

### B. *Features*

- 65 channels to acquire and track satellites simultaneously
- Industry-leading TTFF speed
- Tracking sensitivity reaches 161 dBm
- 0.5 PPM TCXO for quick cold start
- Integral LNA with low power control
- SBAS (WAAS/EGNOS) capable
- Cold start = 29 sec under clear Sky
- Hot start = 1 sec under clear Sky
- Accuracy 5m CEP
- Operable at 3.6V-6V
- Both of RS232 and UART interface at CMOS level
- Small form factor of 32 mm W x 32 mm Lx 8 mm H
- Mountable without solder process
- 6 pins wafer connector

### C. *Serial Port Settings*

The default configuration within the standard GPS firmware is:

- Standard configuration of serial port:
- Supporting 4800/9600 baud rate (Default Value : 9600), 8 data bits, no parity, 1 stop bit, no flow Control

### D. *Improved TTFF*

In order to improve the TTFF (Time To First Fix), has been built with the back-up battery (SEIKO) to support the RTC with a back-up power when no system power is available.

### E. *Wireless Network Communication*

Short for **wi**reless **fi**delity and is meant to be used generically when referring to any type of 802.11 network, whether 802.11b, 802.11a, 802.11g, dual-band etc. Wi-fi [05] is a wireless technology that uses radio frequency to transmit data through the air. In focusing on 3G and WiFi, we are ignoring many other technologies that are likely to be important in the wireless Internet such as satellite services, LMDS, MMDS, or other fixed wireless alternatives. We also ignore technologies such as BlueTooth or HomeRF, which have at times been touted as potential rivals to WiFi, at least in home networking environments.5 Moreover, we will not discuss the relationship between various transitional, or ‘‘2.5G’’ mobile technologies such as GPRS or EDGE, nor will we discuss the myriad possibilities for ‘‘4G’’ mobile technologies.6 While all of these are interesting, we have only limited space and our goal is to tease out what we believe are important themes/trends/forces shaping the industry structure for next-generation wireless services, rather than to focus on the technologies themselves. We use 3G and WiFi as shorthand for broad classes of related technologies that have two quite distinct industry origins and histories.

## TESTING AND RESULTS

### A. *Steps Involved in Testing*

**Step 1:** Click start menu , Launch Lab VIEW

**Step 2:** Open the server file of project in located disk drive and change the visa resource file (Port name of aircraft personal computer)

**Step 3:** Open the client file of project in located disk drive and give the IP address of aircraft personal computer

**Step 4:** Select the disk drive to store the parameters data of airplane

**Step 5:** After that, First Run the sever file and then immediately run the client file

**Step 6:** Take the various parameters information or data in the stored disk drive

### B. *Results*

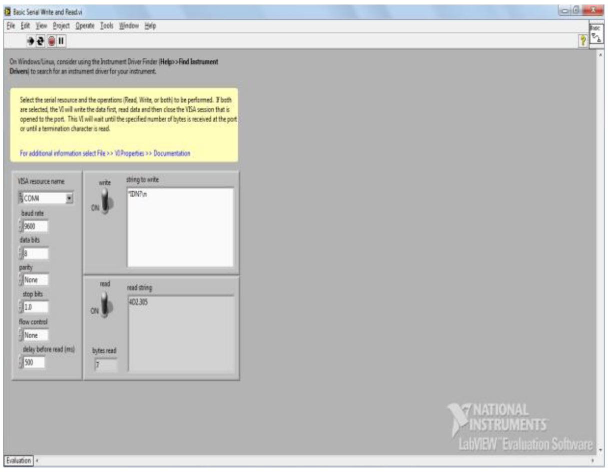


Figure 6 Testing of hardware connection using LabVIEW.

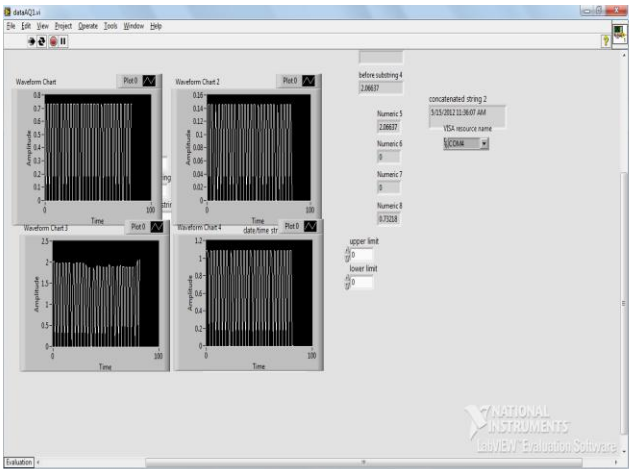


Figure 7 Graphical representation of data from all four ports

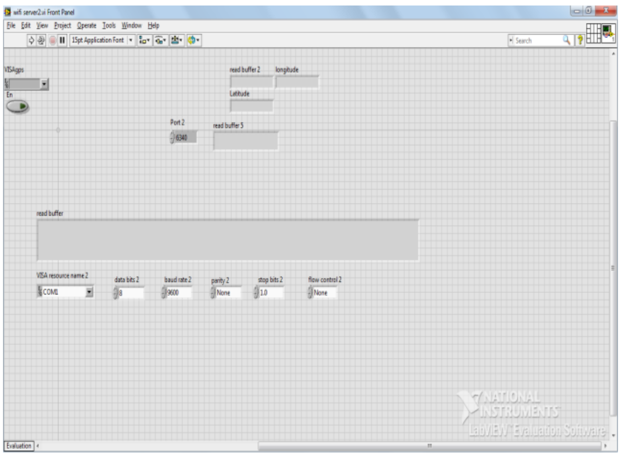


Figure 8 Server front panel

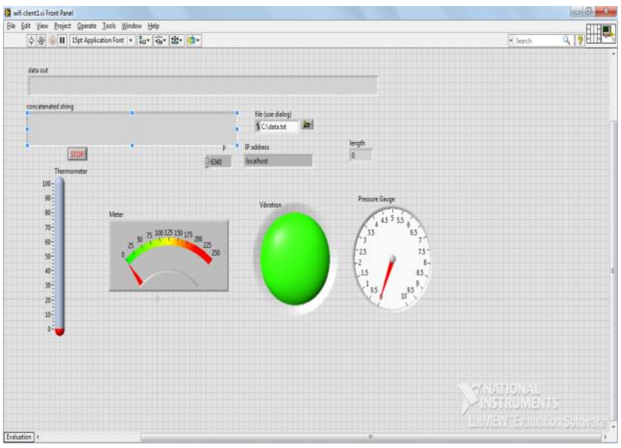


Figure 9 Client front panel

Figure 10 Server front panel when airplane is in normal condition

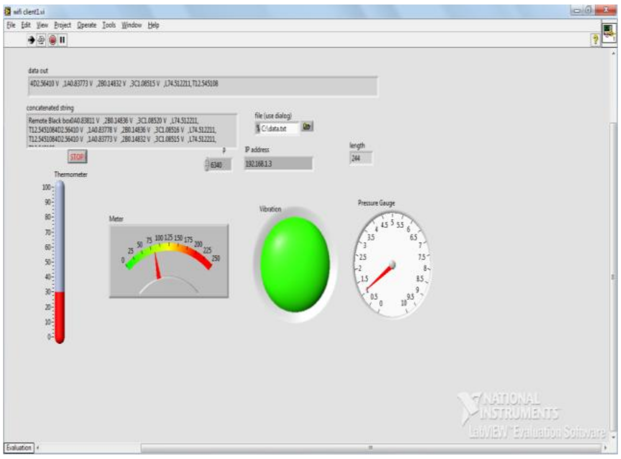


Figure 11 Client front panel when airplane is in normal condition

Figure 12 Server front panel when the airplane is in vibration condition

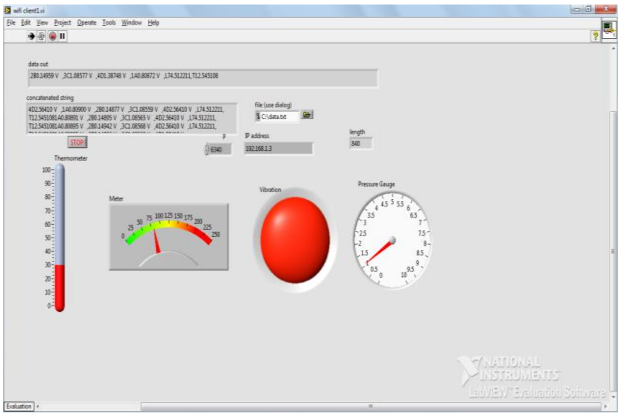


Figure 13 Client front panel when the airplane is in vibration condition

We can easily find location of aircraft and when the accident happened. The entire data should be available in remote pc server, so we need not search for black box. Data will be keep on update every 10 seconds. We can avoid the accident via wireless communication like GSM Zigbee Adhoc etc. GPS is helps to find position velocity and time of aircraft. It helps us to Avoid flight crash, better communication and easy to control.

## CONCLUSIONS

Testing has been carried out extensively using wireless communication technology. The data or information is to be stored in data base of remote PC server and it will continuously updating data to the base station. Results are highly satisfactory as testing has been carried out with different wireless communication technologies GSM and Wi-Fi, the data of each parameter was successful and correct information is displayed. It is required that the database has to be fed with the large number of data to be stored, and it will give the information about exact time, velocity of airplane. However further improvements are required to receive all the parameter information during communication.

.

## REFERENCES

[1] Sandmeier, S. And Itten, K.I: A Physically-Based Model To Correct Atmospheric And Illumination Effects In Optical Satellite Data Of Rugged Terrain. Ieee Transactions On Geoscience Ad Remote Sensing 35, Pp. 708-717.

[2] Holm, M., Lohi, A., Rantasuo, M., Väätäinen, S., Höyhtyä, T., Puumalainen, J., Sarkeala, J. And Sedano, F: Creation Of Large Mosaics Of Airborne Digital Camera Imagery. Proceedings Of The 4th Airborne Remote Sensing Conference And Exhibition, 21-24 June 2007 , Ottawa, Canada.

[3] Pellikka, P.: Development Of Correction Chain For Multispectral Airborne Video Camera Data For Natural Resource Assessment. Fennia 176(1), Pp. 1-110

[4] R. Pallas-Areny and J. G. Webster, Sensors and Signal Conditioning, John Wiley , New York , 1991.

[5] R. Jordan and C.A. Abdallah, “Wireless communications and networking: an overview,” Report, Elect. and Comp. Eng. Dept., Univ. New Mexico, 2002.

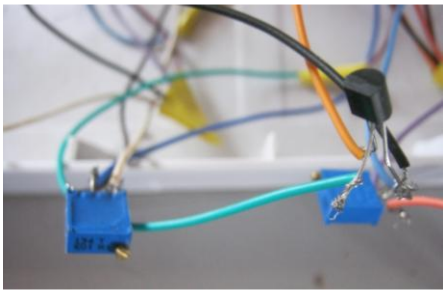
[6] N. Swamy, “Control Algorithms for Networked Control and Communication Systems”, PhD Thesis, Dept. of Elect. Eng., The University of Texas at Arlington, Texas, 2003.

[7] W. Kester, (Ed.), 1992 Amplifier Applications Guide, Section 2, 3, Analog Devices, Inc., 1992, Norwood.

[8] D. Sheingold, (Ed.), Transducer Interfacing Hand book, Analog Devices, Inc., 1980, Norwood.

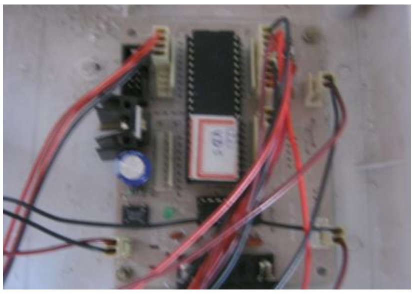
[9] C. Smyth, “ISO 8802/5 token ring local area networks,” Electrical & Communication. Engg. J., vol. 11, no. 4, pp. 195-207, Aug. 1999.

## APPENDIX



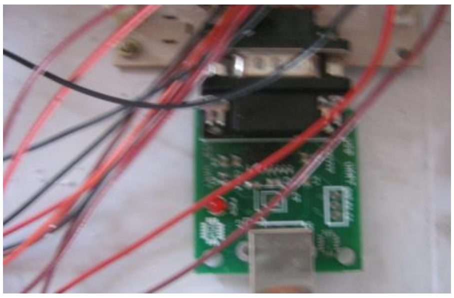
Sensors


Transformer



ATmega32 IC


LCD Display



USB connector

GPS Device

Circuit connections

Circuit box