Fixed Wings UAV Direction Control "Hardware Design"

Mohamed Khalil1, Aisha Abdullah, Sharief F Babikir and Tahani Abdullah

Faculty of Engineering, Neilen University, Khartoum, Sudan
International Islamic University Malaysia, Faculty of engineering, Malaysia, Senior Member IEEE
University of Khartoum, Electronics & Electrical Engineering, Khartoum, Sudan, Senior Member IEEE
University of Khartoum, Electronics & Electrical Engineering, Khartoum, Sudan

Summary
The goal is to design of Unmanned Aerial Vehicle "UAV" system through the control of a stable fixed-wing aircraft to accomplish inexpensive, simple, and still efficient aircrafts by adding an inertial circuitry to the airplane that consists of a GPS unit, a microcontroller unit, and other devices. The aircraft has the ability to navigate from one place to another predefined place.
Additional feature is to switch between two the modes of control: autopilot and manual is provided. In autopilot mode the aircraft is fully controlled by an on-board microcontroller, the only data needed for navigation is to specify the target waypoint, and then the system will navigate fully automatic towards the target until it reaches the target and does the specified duty.
Key words:
UAV, Servo motor, GPS.

1. Introduction

Planes have a very serious weakness: they must be piloted by human beings; that is to say, human beings must be aboard them. One must avoid both the purely negative consequences of crashes (death and property destruction), and the consequences that are positive for one's enemies (the taking of prisoners of war, hostages and sensitive information).
Today, computers with significant computational capabilities continually become less expensive. The economics of consumer electronics has also made inexpensive GPS units available. A remarkable notion is to develop a guidance and flight control architecture that utilizes a single antenna GPS as a sole sensor, resulting in a very low cost system, still capable of rudimentary waypoint navigation, altitude control, and approach courses. The goal of the project is to stabilize and control a small fixed-wing aircraft through the design of an inexpensive, simple, and autonomous Unmanned Aerial Vehicle (UAV) control system.

2. Theoretical Consideration

2.1 Microcontrollers

A microcontroller (also microcontroller unit, MCU) is a small computer on a single integrated circuit; its function is determined by a program loaded in it. Like all computers microcontrollers are equipped with a central processing unit or CPU, a memory system, an input/output system, a clock or timing system, and a bus system to interconnect constituent systems. The bus system consists of an address bus, a data bus, and a control bus.
In Figure (1) we have provided the block diagram of a generic microcontroller. We would like to emphasize that all systems shown in the diagram are contained within the confines of a single integrated circuit package. Each system will be discussed briefly in a clockwise fashion beginning with the memory system [1].

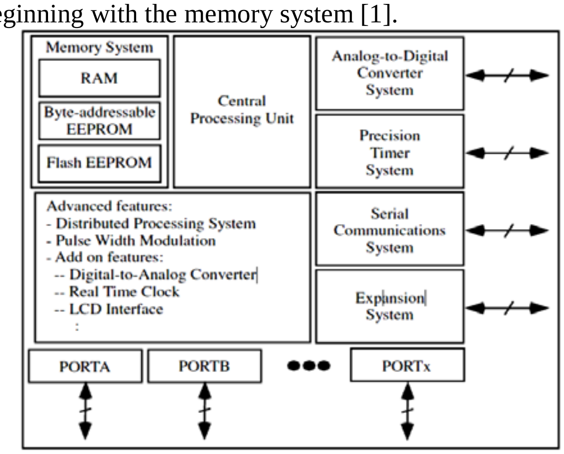

Figure 1: Microcontroller block diagram

2.1.1. Memory System

As its name implies, the memory system contained within a microcontroller is used to remember the algorithm executed by the microcontroller, key program variables, and also system information.
A microcontroller's memory system is usually a conglomeration of different memory technologies. Most

Manuscript received August 5, 2015
Manuscript revised August 20, 2015

microcontrollers are equipped with a memory system containing both random access memory (RAM) and read-only memory (ROM) components [1].

2.1.1.1. RAM

Stand of "Random Access Memory". RAM configurations are used to hold program variables that might change during program execution [1].

2.1.1.2. ROM

Stand of "Read Only Memory". ROM configurations are non-volatile, which makes them an ideal location to store a main program. That way should the microcontroller lose power, it will not lose its main program [1].

2.1.1.3. EEPROM

The EEPROM, or electrically erasable programmable ROM, is available in two different varieties byte-addressable EEPROM and flash EEPROM. Most microcontrollers are equipped with both types. Byte-addressable EEPROM, as its name implies, allows modification of single bytes of information during program execution. This type of memory is useful for storing program constants, security combinations, and fault status. Flash EEPROM may be rewritten in bulk. It does not allow for updating a single memory location. Flash EEPROM is used to store the microcontroller's algorithm [1].

2.1.2. Central Processing Unit

The heart of the microcontroller is the central processing unit or CPU. The CPU contains two main component parts: the arithmetic logic unit (ALU) and the control unit. The ALU performs the arithmetic operations (addition, subtraction, shift right, etc.) and logic operations (AND, OR, exclusive-OR, etc.) for the microcontroller [1].

2.1.3. Crystal Time Base

The time base for the processor is usually provided by a quartz crystal or a ceramic resonator. The quartz crystal provides a more accurate, stable time base [1].

2.1.4. Analog-to-Digital Converter

Most microcontrollers are equipped with multi-channel analog-to-digital converters (ADCs). The analog input signals are converted to a weighted binary representation as shown in Figure 2

To convert an analog sample to a weighted binary value, three steps must be performed: determining the sample rate, determining the required resolution of the converter, and encoding the voltage sample into a weighted binary value [1].

2.1.4.1. Sample Rate

The Nyquist criterion indicates that the analog sample must be sampled at a rate that is at least twice the highest frequency in the sampled signal. This can be expressed as follow [1]:

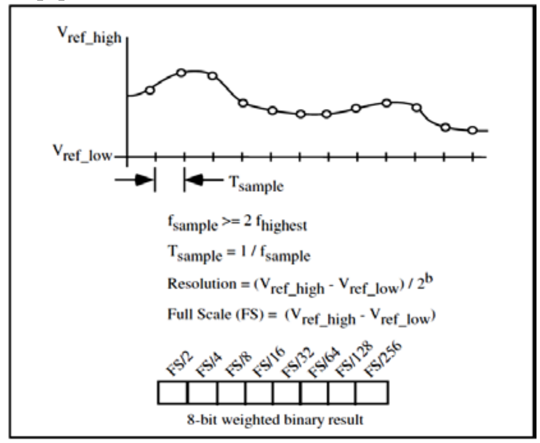

Figure 2: Analog-to-digital conversion

2.1.4.2. Resolution

The equation which ties the different resolution factors together can be expressed as:

$$\text{Resolution} = (V_{\text{ref\_high}} - V_{\text{ref\_low}})/2^{b} \qquad (1)$$

In the equation Vref_high and Vref_low are the reference voltages provided to the ADC. The input analog signal must lie between these two reference values. External conditioning electronics may be required to ensure that this condition is met. The variable b is the number of bits of resolution provided by the ADC [1].

2.1.4.3. Encoding

Full scale voltage is defined as:

$$\text{Full scale voltage} = V_{\text{ref\_high}} - V_{\text{ref\_low}} \qquad (2)$$

To convert the weighted binary value to a floating-point (real number) representation, the following conversion may be used [1]

$$voltage = \left(weighted\ binary\,\frac{input}{2^{b}}\right)(full\ scale\ voltage) \qquad (3)$$

2.1.5. Timing system

Most microcontrollers are equipped with a multi-channel precision timing system. The timing system has a variety of precision timer functions including measuring the parameters of an incoming digital signal, generating a precision output signal, or counting events [1].

2.1.5.1. Measuring parameters of input signals

The key parameters of an input signal that may be measured are period, frequency, and duty cycle. The parameters are measured by configuring a timer channel to log the count of the free running counter when certain signal parameters (rising and falling edges) occur [1].

2.1.5.2. Counting events

A precision timing system may also be used to count events. For example, if we equipped a rotating motor with an encoder that provided "n" number of pulses per motor revolution, we could develop an algorithm to determine motor speed in revolutions per minute (rpm) [1].

2.1.6. Serial communications system

All microcontrollers are equipped with one or more serial ports to communicate with external devices. Typical microcontrollers have two different types of serial communication subsystems on board: one or more asynchronous communication systems and one or more synchronous communication systems. The first type briefly discussed the next [1].

2.1.7. Asynchronous communications

As the name indicates, an asynchronous communication system uses a stringent protocol to communicate with other serial communication systems. The most widely used asynchronous communication technique is the RS-232D (RS stands for Remote Standard) interface.
Data is transferred using the ASCII (American Standard Code for Information Interchange) standard or the newer international Unicode coding standard between two serial communications equipped systems.
The serial communication can be performed in the simplex mode, which allows one direction of communication at a time, or the duplex mode, which allows two-way communication simultaneously. Microcontrollers may use the duplex mode to interface with external devices. To protect the integrity of the data transferred, the software protocol of the RS-232D method requires data to be transferred in a frame that contains data bits (8 or 9 bits), a start bit, and a stop bit. It also specifies the communication rate, which is called the baud rate (bits per second), to ensure that the bit transmit rate matches with the bit receive rate [1].

3. TOOLs

3.1 PIC16F877A Microcontroller

The PIC16F877A microcontroller is one of a family of Harvard architecture microcontrollers made by Microchip Technology. It features up to 200 ns instruction execution, 256 bytes of EEPROM data memory, self-programming, an ICD, 2 Comparators, 8 channels of 10-bit Analog-to-Digital converter, 2 capture/compare/PWM functions, 3 timers, a synchronous serial port that can be configured as either 3-wire SPI or 2-wire I2C bus, a USART, and a Parallel Slave Port. These features of high memory capacity and fast execution rate are behind choosing this microcontroller [2].

3.2 PIC16F87 Microcontroller

The PIC16F877A microcontroller is one of a family of Harvard architecture microcontrollers made by Microchip Technology. It belongs to the Mid-Range family of the PICmicro devices. It has wide operating voltage range: 2.0V to 5.5V, 10-bit, 7-channel Analog-to-Digital Converter, Addressable Universal Synchronous Asynchronous Receiver Transmitter (AUSART/SCI) with 9-bit address detection RS-232 operation using internal oscillator (no external crystal required) [2].

3.3 Mobile platform

The UAV system mobile platform is a fixed wing small plane which must have the following specifications:

- Simple: simplicity in its internal construction, and can be controlled easily.
- Inexpensive.
- Has the ability to add an additional load (autopilot).

Regarding to the above requirements and depending on the design principles the following one was chosen: The trading name: **Super Cub EP park flyer** from Thunder Tiger Co [3].

3.4. Servo motor

A Servo is a small device that has an output shaft. This shaft can be positioned to specific angular positions by sending the servo a coded signal. In practice, servos are used in radio controlled airplanes to position control surfaces like the elevators and rudders. They are also used in radio controlled cars and of course, robots.
The servo motor were chosen is Tower Pro SG-5010 because it has a light weight and works with 5V like all other components used in the design so we don't have to use any kind of voltage regulator [4].

Figure 3: Tower-pro typical servo motor

3.4.1. Specifications:

- Weight: 1.44oz / 39g (Servo Net Weight Only)
- Temperature Range: -30 to +60 Degree C
- Dead Band Width: 4µsec
- Operation Voltage: 4.8 - 6 Volts
- Stall Torque (6.0V): (6kg/cm).

3.5 GPS receiver

There were many options that were looked at when finding the most feasible GPS receiver.
The required GPS receiver must has the following specifications

- Light weight.
- Fast positioning acquisition.
- Low power consumption.
- High accuracy.

Regarding on the above requirements, and depending on design principles, the chosen GPS receiver is: GARMIN GPS 18-5Hz.
GARMIN GPS 18-5 receiver module has additional features over that mentioned above:
This GPS receiver was chosen because of:

- Very light weight (162 grams).
- Interface to a serial port, this unit accept RS-232 level inputs and transmit voltage levels that swing from ground to positive supply voltage.
- It needs 5V to operate, so it can be connected to the same power supply of the microcontroller.
- The company provides software to select the required format of the output sentences and to select the desired baud rate (so it is able to work with the same baud rate of the microcontroller (9600 bps)).
- It is very accurate (less than 3 meters error) which is needed when doing the tests in small area.
- The received data is updates 5 times per second and it can be in many formats [5].

3.6. MAX232 Level Converter

The HIN232 (Figure 4) is one of a family of RS-232 transmitters/receivers interface circuits that meet all ElA RS-232E and V.28 specifications, and are particularly suited for those applications where ±12V is not available. They require a single +5V power supply and feature onboard charge pump voltage converters which generate +12V and -12V supplies from the 5V supply [6].

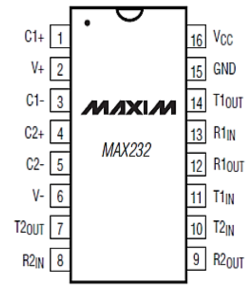

Figure 4: MAX232 IC layout

3.7. 24FC515 serial EEPROM

The Microchip Technology Inc. 24FC515 is a 64K x 8 (512K bit) Serial Electrically Erasable PROM, capable of operation across a broad voltage range (1.8V to 5.5V). It has been developed for advanced, low power applications such as personal communications or data acquisition. This device has both byte write and page write capability of up to 64 bytes of data. This device is capable of both random and sequential reads [7].

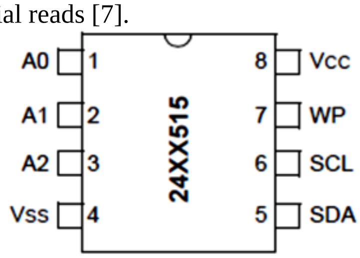

Figure 5: 24FC515 EEPROM IC layout

3.8. ULN 2003

The ULN2003 is a monolithic high voltage and high current Darlington transistor arrays. It consists of seven NPN Darlington pairs that features high-voltage outputs with common-cathode clamp diode for switching inductive loads. The collector-current rating of a single Darlington pair is 500mA. The ULN2003 has a 2.7kΩ series base resistor for each Darlington pair for operation directly with TTL or 5V CMOS devices [8].
FEATURES:

- 500mA rated collector current(Single output)
- High-voltage outputs: 50V
- Inputs compatible with various types of logic.
- Relay driver application.

4. DESIGN

4.1. Plan of work

Step 1: System design: This includes:

- Developing the general system design with respect to the functional and non-functional requirements.
- Design the individual subsystems such as positioning and data acquisition subsystems and determine the appropriate interfaces between them.

Step 2: components ordering: this includes:

- Determine the components required to achieve each subsystem.
- Order these components

Step 3: Putting the GPS to work: This includes:

- Make the necessary connections to connect the GPS to the computer.
- Configure the GPS to output GPRMC sentences only and to operate with baud rate 9600 Kbps.
- Test the GPS by the computer and ensure that it works as configured.

Step 4: Test the servo motor: This includes:

- Test the servo motor using oscilloscope and signal generator to verify servo characteristics (duty cycle, signal amplitude, etc…).

Step 5: Interfacing GPS to the Microcontroller: This includes:

- Implement necessary connections to interface the GPS to the positioning microcontroller through MAX232.
- Develop the navigation software and load it into the major microcontroller.
- Test the connection to the MAX-232 and the navigation software using LCD.

Step 6: Interfacing the servo motor to the microcontroller: This includes:

- Connecting the servo motor to the major microcontroller through ULN2003 Darlington array chip.
- Develop controlling program and load it into the secondary microcontroller.

Step 7: Simulation: This includes

- Simulation software to construct the circuit.
- Implement virtual serial interface to give arbitrary positions to the system to ensure both software's are working well when connecting the two microcontrollers together and the system is working as expected.

Step 8: System integration:

- Connect the subsystems described in steps 5 and 6 which is the complete UAV system.
- Connect an LCD with positioning microcontroller and an oscilloscope with controlling microcontroller.
- Move the GPS and notice the change in the angle on the LCD and the PWM signal on the oscilloscope.
- Instruct the UAV system to go to specific locations and make a tour to ensure the system is working well.
- By the end of this step the UAV system is complete.

4.2. High Level Design

By high level design we mean to see the project from an overall view point, then to get down and consider the project as a system of main parts.

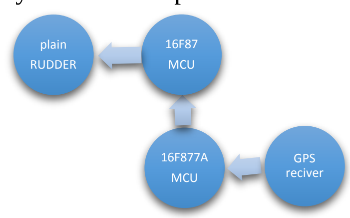

Figure 6: Overall view of the project

Unmanned Air craft system consists of two control units (two MCUs) and single controlled element (plane rudder) as shown in Figure 5.The function of the first MCU is to collect the GPS sentence and make the heading calculations; while the second MCU controls the plane rudder according to the calculations from the first MCU.

4.2.1. Heading calculations

The PIC16F877A performs the majority of the calculations for waypoint navigation. That is, it reads in the latitude and longitude data from the GPS receiver and then calculates the current aircraft heading and the heading to the next waypoint as shown in figure 6.

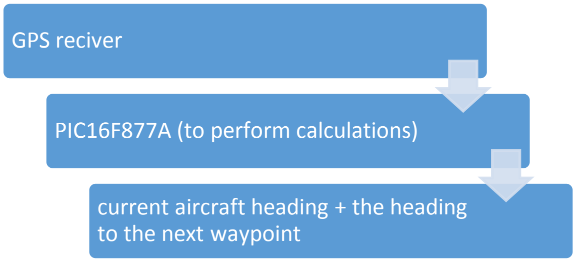

Figure 7: heading calculations

4.2.1.1. Rudder control

Once the direction and amount of turn is known, the PIC16F877A communicates this information to the PIC16F87 which is used to deflect the rudder servo accordingly. The PIC16F87 is also used to perform manual override of the aircraft via the RC transmitter. As shown in figure 7.

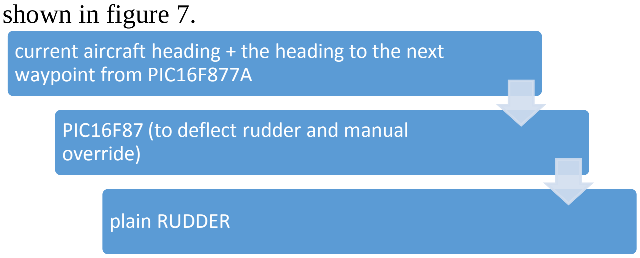

Figure 8: Rudder control

In the event that the algorithm is not performing as expected (or if the plane is about to crash); a flick of the transmitter switch will allow a human pilot to take over and avoid the crash.

4.2.1.2. Detailed Project Design

The previous section has given a brief and fast idea about the project design as whole, this section will go further to the detailed design of both the hardware design and the software design.
As illustrated in Figure 8, hardware design is classified into two categories: detailed circuit designs, and the entire circuit layout design. On the other hand, the software design is classified into two categories: 16F877A and 16F87 microcontroller's C codes.

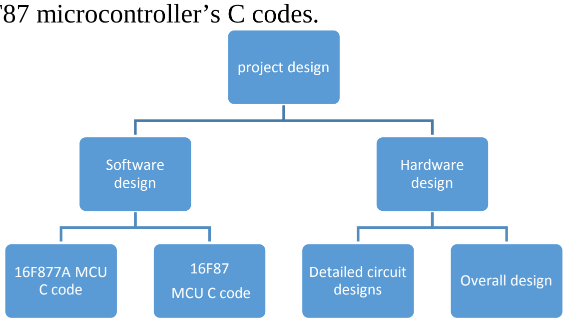

Figure 9: Project design division

4.3. Hardware design

4.3.1. GPS interface

➢ **GPS to PIC16F877A connection**

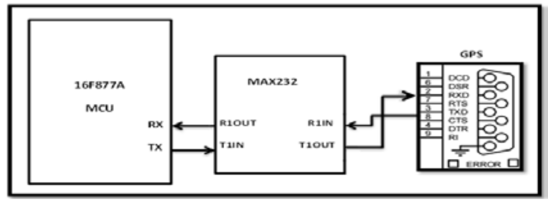

Figure 10: GPS interface

To overcome the difference in voltage between the GPS and the microcontroller, MAX-232 where used as an interference that converts signals from the GPS receiver to signals suitable for use in TTL compatible digital logic circuits. The MAX232 is a dual driver/receiver and typically converts the RX, TX, CTS and RTS signals as shown in figure 9.

4.3.2. Servo motor interface

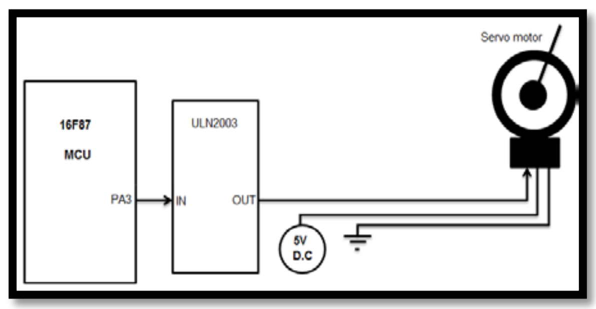

Figure 101: Servo motor interface

4.3.3. EEPROM interface

The original plan was to fly the plane manually over certain waypoints and every time a waypoint to be selected, It must quickly switch from manual to autonomous control and then back to manual (using the plane remote control). This would signal to the microcontroller that to store this point as a waypoint. This would eliminate the need to enter these in manually. However, no enough memory on the PIC16F877A microcontroller to create variables for these waypoints to be stored in, so the EEPROM where used as secondary storage. Figure (10) illustrates the interface.

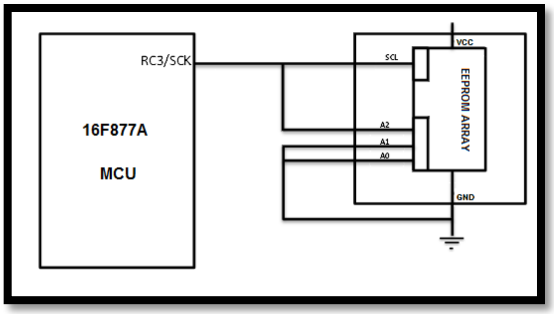

Figure 12: EEPROM interface

4.3.3.1. Two microcontrollers interface

The two microcontrollers (16F877A and 16F87) are connected as shown below, the PIC 16F877A calculates the current aircraft heading and the heading to the next waypoint and send these information as a signal to PIC16F87 to control the rudder servo as shown in sections above.

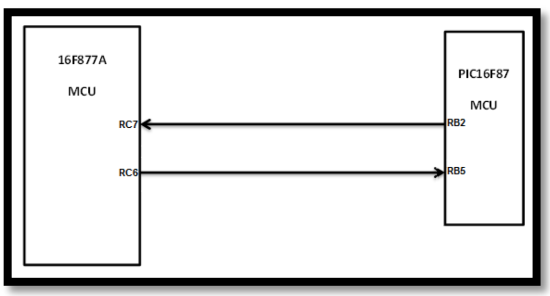

Figure 13: PIC16F877A and PIC16F87 microcontroller's connection

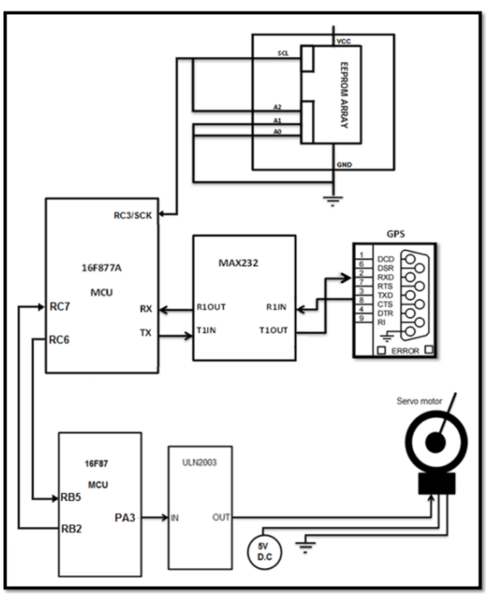

Figure 14: Complete design of UAV circuit

4.3.3.2. Complete Circuit Layout

By complete circuit layout, it is meant the integration of all above parts into one scheme so as to illustrate the ultimate design of unmanned aircraft vehicle as shown in Figure (12)

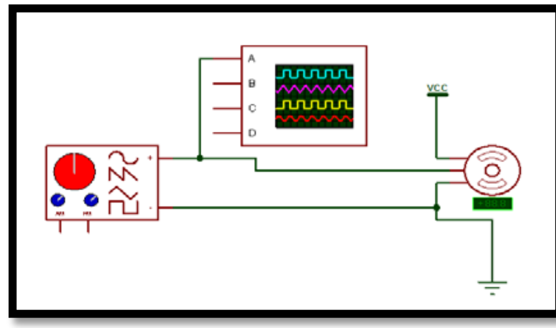

Figure 1115: Servo motor simulation circuit

4.4. System tests

4.4.1. Servo motor test

To determine servo's characteristics the servo was first connected with a signal generator in order to determine its period which was founded 20 ms. Also its maximum duty cycle in order to turn in both directions was measured.
The real hardware implementation has been done successfully by connecting the previous circuit, the same results observed in the lab for tower-pro servo motor.

4.4.2. GPS test

The GPS was connected to a computer through serial port and the sensor configuration software SNSRCFG_320 was used to configure the GPS to GPRMC sentence only and disable any other sentences. Also it was used to configure the GPS to work with baud rate equal to 9600 bps as the same as the microcontrollers receives.

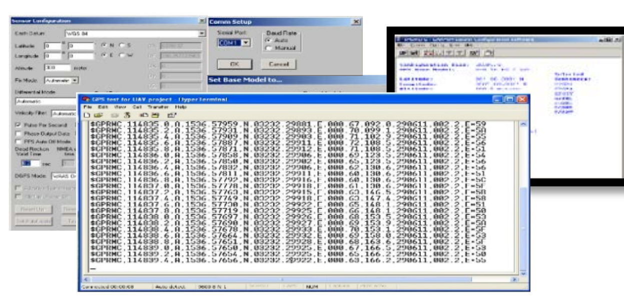

Figure 16: GPS configuration

Then the GPS was connected with the hyper terminal, to verify the GPS produce the desired sentences only and work as expected.

4.4.3. Servo to PIC 16F87 through ULN 2003 interface

PIC-16F87 was programmed to produce square signals with different duty cycles to ensure this signal can be supplied to the motor through ULN 2003 which also drive the servo motor with sufficient current.

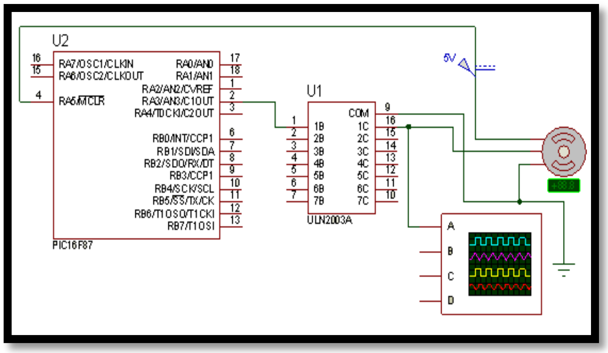

Figure 1712: servo motor to PIC interface

4.4.4. GPS interface

- The GPS was connected to PIC-16F877A through MAX 232
- An LCD was connected with PIC-16F877A to ensure MAX 232 was connected correctly and PIC-16F877A program is working well and extract the desired data from the GPS. Also, this circuit verified that the GPS is working as expected.

4.4.5. Constructing the whole circuit

The last step was to connect all the subsystems

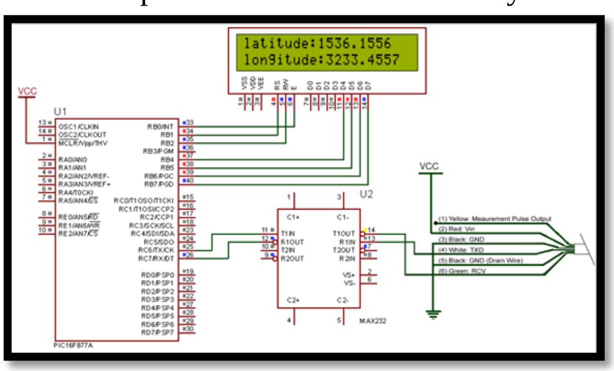

Figure 18: GPS to PIC interface

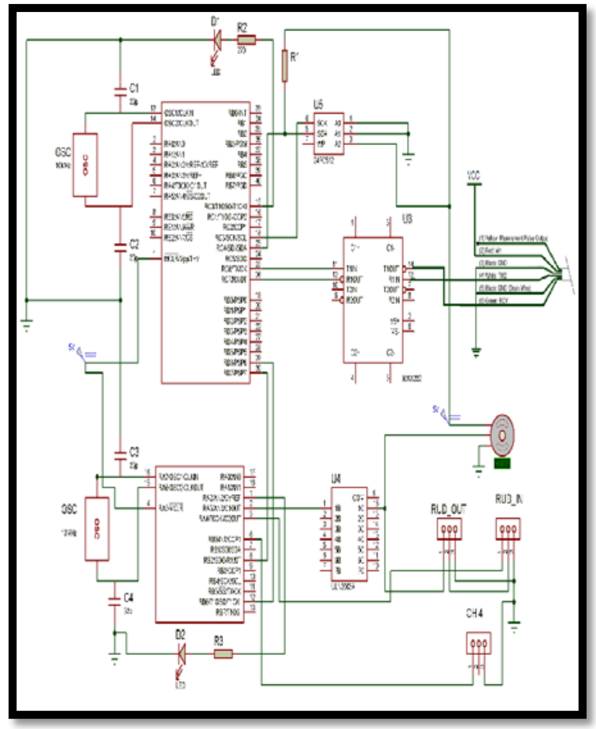

Figure 19: complete circuit

above to produce the final circuit which looks like above figure.

5. Conclusion

The whole system was tested by using an LCD and LEDs as follow:

- To verify the power was supplied to the circuit, two LEDs were used and the programs of the microcontrollers were configured to turn the LEDs on for 1.5ms and then turn them off.
- To verify that the system can be switched from manual to auto mode and vice versa, a LED connected to PIC-16F87, and the program of the microcontroller was configured to switch the LED on when working in the auto mode and switch it of when working in the manual mode. By toggling the transmitter's switch this verifies that the system can be switch from auto to manual mode and vice versa.

To test PIC-16F877A program and verify that it extract the required data from the GPS and makes the calculations correctly, an LCD was used to display the amount of the angle that the plane should turn. And by instructing the system to go to different locations and making a tour to those locations, PIC-16F877A program was verified.

- To test PIC-16F87 program and verify it generates the required signal to control the servo according to the data received from the positioning microcontroller, an oscilloscope was used and connected to the microcontroller, and the was moved from one place to another. This verified that the duty cycle changes as the data received from the positioning microcontroller change.

References

[1] Tim Wilmshurst, “Designing Embedded Systems with PIC Microcontrollers Principles and applications”, First edition 2007
[2] “PIC16F87XA Data Sheet”, 28/40/44-Pin Enhanced Flash Microcontrollers, Microchip Technology, 2003.
[3] Thunder tiger , “Super Cub EP park flyer, thunder tiger2007.
[4] URL“http//:www.servodatabase.`com/servo/towerpro/sg-50 10”
[5] Garmin, “ GPS 18 technical specifications”, Garmin international Inc.
[6] “MAX232 Level Converter Datasheet”, MAXIM Corporation, February 11 2003.
[7] “CMOS Serial EEPROM”, Microchip Technology 2005.
[8] Linear integrated circuit, “ULN2003”, YOUW ANG ELECTRONICS.