



Islamic University of Technology

**Design and Analysis of Tracking Device
for Wireless Embedded System**

A Thesis Presented to

The Academic Faculty

By

Md.Tausif Uddin, Student ID: 102435

Md.Fahim Rahman, Student ID: 102447

Saleh Farhad, Student ID: 102448

Maisra Abdul Azim Zaki, Student ID:092479

In Partial Fulfillment

Of the Requirement for the Degree

Bachelor of Science in Electrical and Electronic Engineering

Academic Year: 2013-2014

A Dissertation on,

Design and Analysis of Tracking Device
for Wireless Embedded System

Submitted By

Md.Tausif Uddin (102435)

Md.Fahim Rahman (102447)

Saleh Farhad (102448)

Maisra Abdul Azim Zaki(092479)

Approved By

Dr. Md. ShahidUllah
Head of the Department& Professor
Department of EEE, IUT

Dr. Khandokar Habibul Kabir
Thesis Supervisor & Associate Professor
Department of EEE, IUT

ABSTRACT

A large number of children are being kidnapped on a regular basis which is a matter of great anxiety for the parents. Then again a huge number of people who are sufferers of Alzheimer disease or mental degradation are reported missing every year. Thus, in order to ensure the security of life for these particular individuals, we would require a precise tracking device at a low cost. The purpose of this project is to use the global positioning system (GPS) and the Global System for Mobile Communication (GSM) for tracking purposes using Google Maps. The system is divided into two parts, which consists of the tracking part and the displaying part. For the tracking part, we used a microcontroller along with a gps and gsm module, which altogether makes the embedded system. The GPS module provides the information of the location of the subject to the microcontroller. Then the GSM module sends the co-ordinates of the location to the user. The data is sent through Short Message Service(SMS) by the GSM module. And the message is received in a cellular device. Then the co-ordinates in the cellular device are matched with the co-ordinates in Google maps. Then the information about the subjects location is obtained. in. In this project, we used a GSM module with a SIM card. This SIM card sends the data in the form of a SMS to the receiver. Then the data is matched with Google maps by the cell-phone. In the modern world this type of tracking devices are available, but they come at a high cost. So our main goal was to make the tracking cost efficient. To achieve our goal we first established that our device can track precisely and then we gave our views about the cost minimization.

Content

Chapter: 1	Introduction	6
1.1	Project Background.....	6
1.2	Problem Statement.....	6
1.3	Objectives.....	6
1.4	Scope of Work.....	7
1.5	Thesis Outline.....	7
Chapter:2	Wireless Tracking Device	8
2.1	Introduction.....	8
2.2	What is a tracking device.....	8
2.3	Global Positioning Device(GPS).....	9
2.4	Global System for Mobile Communication (GSM).....	13
2.5	Related Works.....	15
Chapter: 3	Design of Tracking Device	19
3.1	System Overview.....	19
3.2	Design.....	20
3.3	Development of tracking device.....	24
3.4	Coding.....	26

Chapter: 4	Tracking Data Analysis	27
4.1	Tracking Results.....	27
4.2	Cost Analysis.....	31
Chapter: 5	Conclusions and Recommendations	55
5.1	Conclusion.....	55
5.2	Recommendations.....	55
References		56
Appendix		58

CHAPTER-1

INTRODUCTION

1.1 Project Background

The wide coverage of cellular and satellite network leads to various useful applications that increase the convenience of our daily life. One of such applications is people tracking or positioning system. The location of a person can be tracked by using the GPS (Global Positioning System) and GSM (Global System for Mobile communication) technology.

GPS is a satellite based navigation system which can provide accurate location, time, speed and direction data. The satellite that is placed into the orbit will transmit the coded information to the GPS receiver. This data allow us to determine the precise location on the earth by measuring the distance from the satellite.

GSM is the digital mobile network which is used to transmit the mobile voice and data service through the narrow band with Time Division Multiple Access (TDMA) Technique. It can used to detect the location of the mobile phone based on the signal strength to the nearby antenna tower. The GSM technology operates in several bands of frequency for different country. For Malaysia, GSM is set to use 900 MHz and 1800 MHz frequency bands.

1.2 Problem Statement

Various Tracking devices are available nowadays. But they are costly. So, people of the middle-class families cannot afford them. So children and old people suffering from mental degradation from those families are more insecure. So we would suggest a cost efficient solution to this problem

1.3 Objectives

The objectives are:

- Design and analyze a tracking device
- Use the device for tracking
- Match the tracking data with Google maps
- Cost comparison and suggesting a cost effective solution

1.4 Scope of Work

The main emphasize of this project is to design a cost efficient people tracking embedded system which involves both hardware and software part. The scope of this project is as follows:

- (i) To implement the cost efficient solution and suggest further solutions
- (ii) To implement the GSM modem for receiving the information from the GPS and sending the message to user.
- (iii) To create a user interface for displaying the current location of the user.
- (iv) To integrate real time tracking cost efficiently.

1.5 Thesis Outline

This thesis consists of five chapters. First chapter focuses on the project background, problem statements, objectives and scopes of the project. Chapter two will review all of the related study regarding on this project. In Chapter three, the discussion will be on the methodology in conducting this project which includes the methods and techniques used. In Chapter four, the results and the discussion will be discussed. Finally, conclusion and future work are presented in chapter five.

CHAPTER-2

WIRELESS TRACKING DEVICE

2.1 Introduction

The main objective of this project is to design a a people tracking embedded system by using GPS and GSM modem and compare its cost with the market prices of other tracking devices and hence provide a cost efficient solution to the problem.. This chapter will discuss about the basic concept of the technologies involved. Besides that, some of the related works will be discussed as well.

2.2 What is a Tracking Device?

A tracking device is an electronic tag that can be used to monitor the location of an object or people by using the radio signal or satellite signal. Basically, the design of the tracking devices is depending on several factors such as the nature of the object being tracked, the information needed by the tracker and the budget of the tracker. The indoor location tracking is available with various technologies such as ultrasonic, mechanical, infrared, inertial or radio signal measurement. The GPS is the most effective outdoor tracking system with high accuracy.

The tracking devices can be categorized into two groups which are globally and locally. For local tracking system. It does not require the global coverage such as GPS. It operates based on the local technology such as Wi-Fi, Bluetooth and RFID. While for global positioning system, satellites have to be used in order to obtain the required information.

Besides that, the tracking system can also be divided into passive or active device. The active tracking devices will send out a constant signal continuously while the passive tracking system only will send out a signal when the user require the data.

2.3 Global Positioning System (GPS)

Navigation System with Timing and Ranging Global Positioning System, NAVSTAR is the full description of GPS. GPS [1] is a space based navigation system which being developed, operated and maintained by the Defense Department of US. Although it is being controlled by the US government, it is freely accessible by anyone with a GPS receiver.

The GPS is comprised of three important segments as shown in Figure2.1. The segments of GPS are:

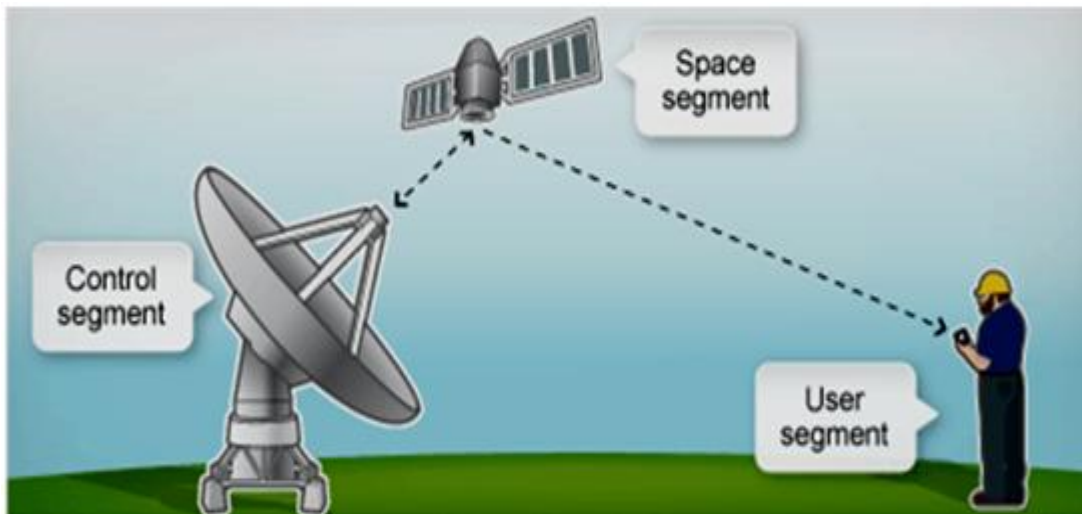


Figure2.1: Segments of GPS

- (i) **Space segment:** The space segment consists of satellites that orbit the earth on six different orbital planes. Each of these planes has four satellites which will transmit the one way signals to the receiver equipment on earth.

- (ii) **Control segment:** The control segment is the earth equipment that carrying out the task of monitoring and controlling the space segment, satellite tracking, telemetry and maintain the satellite orbit configuration.

- (iii) **User segment:** The user segment is the satellite receiver equipments which are used to receive the signal from the satellites and determine the current location of the user based on the received signals.

The GPS is suitable to be used in the people tracking embedded system because the GPS can work in any weather condition, anywhere in the world, 24 hours a day with no subscription fees or setup charges [2]. Besides that, the GPS also can provide 3 dimensional positioning. Thus, it can be used to detect the location of the user with high accuracy.

- **GPS Position Determination**

The satellite of GPS will transmit the one way signals to the GPS receiver equipment on the earth. Every satellite will transmit the data that indicates its location and the time they sent out the signal. The timing information plays an important role in determining the users location on the earth, thus, GPS satellites are equipped with atomic clock on board to provide an accurate time reference[3]. The distance between the particular satellite and the GPS receiver as shown in Figure2.2 can be determined by calculating the travel time of a signal from the satellite to the receiver, where:

Travel time = signal reception time - signal transmission time

Distance = travel time * speed of light

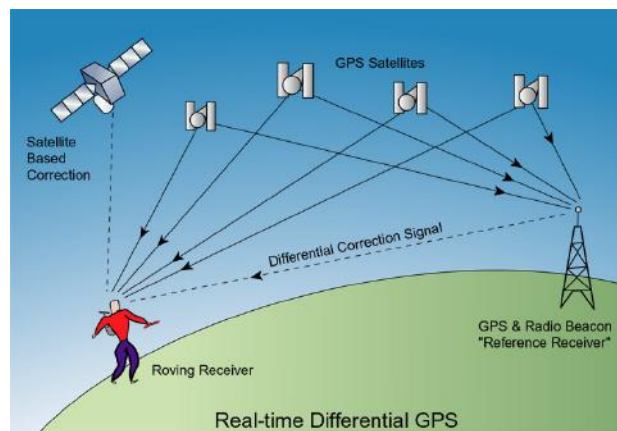


Figure2.2: The distance between satellite and one's position on earth

Trilateration (triangulation) is used to calculate the current position of the GPS receiver based the information on GPS signal's travel time from three nearby satellites and their exact locations in the orbit. However, in order to determine one's location in 3Dspace,4 satellites are needed instead of three[4]. Figure2.3 shows the intersection point that indicates the location of GPS receiver.

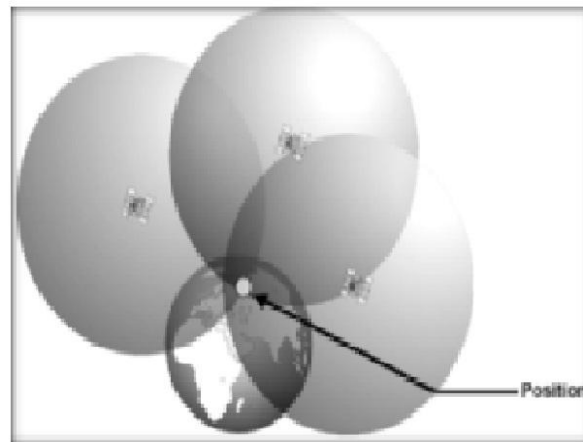


Figure2.3: The intersection Point indicates the location of the GPS receiver

- **NMEA Data**

The full description of NMEA is National Marine Electronics Association. This association has developed a specification that is used to define the interface between various pieces of marine electronic equipment which includes the GPS receiver communication. Each type of devices has their own standard sentences and each of these sentences has two letter prefix. For example, the prefix for GPS receiver is GP. There are another three letters followed by prefix that are used to define the contents of the sentences. The standard sentence is begins with \$ sign, followed with “talker ID” (2 characters), “message ID” (3 characters), various data fields(each information is separated by commas), optional checksum (begins with * sign), and ends with carriage return or line feed[5]. Figure2.4shows the list of GPxxx sentences.

<u>\$GPBOD</u>	- Bearing, origin to destination
<u>\$GPBWC</u>	- Bearing and distance to waypoint, great circle
<u>\$GPGGA</u>	- Global Positioning System Fix Data
<u>\$GPGLL</u>	- Geographic position, latitude / longitude
<u>\$GPGSA</u>	- GPS DOP and active satellites
<u>\$GPGSV</u>	- GPS Satellites in view
<u>\$GPHDT</u>	- Heading, True
<u>\$GPR00</u>	- List of waypoints in currently active route
<u>\$GPRMA</u>	- Recommended minimum specific Loran-C data
<u>\$GPRMB</u>	- Recommended minimum navigation info
<u>\$GPRMC</u>	- Recommended minimum specific GPS/Transit data
<u>\$GPRTE</u>	- Routes
<u>\$GPTRF</u>	- Transit Fix Data
<u>\$GPSTN</u>	- Multiple Data ID
<u>\$GPVBW</u>	- Dual Ground / Water Speed
<u>\$GPVTG</u>	- Track made good and ground speed
<u>\$GPWPL</u>	- Waypoint location
<u>\$GPXTE</u>	- Cross-track error, measured
<u>\$GPZDA</u>	- Date & Time

Figure2.4: List of GPxxx sentences

- **\$GPGGA Sentence**

\$GPGGA sentence is one of the NMEA sentences which is commonly used. The talker ID “GP” stands for GPS while the message ID “GGA” contains position fix information as shown in Figure 2.5. \$GPGGA sentences contain the information of UTC time, which is the time when the GPS data is taken and followed by latitude, latitude ordinal, longitude, longitude ordinal, GPS fix quality, number of satellites being tracked, horizontal dilution of position, altitude, height of geoid, time in seconds since last DGPS update, DGPS station ID number and checksum data. In order to determine one’s location, there are only four critical information is needed, which are latitude, latitude ordinal, longitude and longitude ordinal. Number of satellites being tracks is depend on how many satellites being used to determine one’s location while the horizontal dilution of position represent the relative accuracy of horizontal position. Besides that, altitude determine the position above mean sea level and the checksum data is used by program to check for transmission errors. By referring to Figure 2.5, the data was taken at 12:35:19, the GPS receiver is located at Latitude 48 deg 07.038’

N and Longitude 11 deg 31.000' E, 545.4 meter above mean sea level and 46.9 meter above WGS84 ellipsoid. Furthermore, 8 satellites being used to track the location of GPS receiver and there is no last update and do not have station id.

```
$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47

Where:
  GGA      Global Positioning System Fix Data
  123519   Fix taken at 12:35:19 UTC
  4807.038,N Latitude 48 deg 07.038' N
  01131.000,E Longitude 11 deg 31.000' E
  1        Fix quality:
           0 = invalid
           1 = GPS fix (SPS)
           2 = DGPS fix
           3 = PPS fix
           4 = Real Time Kinematic
           5 = Float RTK
           6 = estimated(dead reckoning) (2.3 feature)
           7 = Manual input mode
           8 = Simulation mode
  08       Number of satellites being tracked
  0.9      Horizontal dilution of position
  545.4,M  Altitude, Meters, above mean sea level
  46.9,M   Height of geoid (mean sea level) above WGS84
           ellipsoid
  (empty field) time in seconds since last DGPS update
  (empty field) DGPS station ID number
  *47     the checksum data, always begins with *
```

Figure2.5: Explanation of \$GPGGA sentence

2.4 Global System for Mobile Communication (GSM)

GSM is the short for global system for mobile communication [6]. It was a communication system that originated in Finland Europe and developed by using digital technology. GSM is a 2G technology that is implemented globally and used to transmit voice and low volume digital data service. Examples of low volume digital data are SMS (short message service) and MMS (Multimedia Message Service). Besides that, GSM has four frequency ranges, which are 850 MHz, 900MHz, 1800MHz and 1900MHz. In Malaysia, the GSM standard uses the 900MHz and 1800MHz.

- **GSM module**

GSM module is similar to mobile phone. It is a specialized wireless module which needs a SIM card and works with a GSM wireless network. GSM modem utilizes the radio wave for sending and receiving the messages. Utilization of SMS technology has become popular because it is an inexpensive, convenient and accessible ways of transferring and receiving data with high reliability[7]. Besides that, GSM modem can be used for automating business process, sending SMS from a computer and vehicle tracking with integrated GPS.

There are three different types of GSM module, which are:

- (i) A GSM module with SIM card can be an external modem device which is connected to a computer through USB port, serial port, Bluetooth or infrared.
- (ii) A GSM module can be a PC card or PCMCIA card which is installed in a notebook computer.
- (iii) A GSM module can also be a standard GSM mobile phone

A GSM modem is controlled by using the AT commands. If the user would like to do the operation such as reading, writing, deleting and sending messages, an extended set of AT commands which are defined in the GSM standard is needed.

- **GSM Network**

GSM network[8] is a public land mobile network (PLMN). Mobile station which is made up of a SIM (Subscriber Identity Module) card is the user terminal in GSM network while the mobile terminal refers to the user device such as mobile phones. A system that uses a cellular network based around broadcast stations or satellite technology that is connected to signal from orbit are part of the GSM network. The main purpose of the GSM network is to facilitate easier access to cellular and satellite platforms across international lines.

A GSM network comprises of three major systems, which are:

- (i) **Switching system (SS):** The main functions of this system are performing call processing and subscriber related task. The functional units under the switching system are home location register (HLR), mobile services switching center (MSC), visitor location register (VLR), authentication center (AUC) and equipment identity register (EIR).
- (ii) **Base station system (BSS):** Base station system is responsible for radio related functions, which consists of the base station controller (BSCs) and base transceiver station (BTSs). The base station controller is used to manage the resource distribution while the base transceiver station is radio equipment which responsible for handling the radio interfaces to the mobile station.
- (iii) **Operation and support system (OSS):** The OSS will be connected to the equipment in the switching system and base station system. Operation and support system is used to provide a network overview, customer support for operation and maintenance activity which required for a GSM network.

2.5 Related works

There are various technologies that can be used to detect the location of an object based on the requirement of the user. The differences between these technologies include the cost needed, accuracy, availability, type of coverage and coverage area. An overview will be given on some previous works which are related to tracking.

- **Application of GSM and GPS Technology**

The wide coverage of cellular and satellite network leads to various useful applications such as vehicle tracking system, agriculture monitoring system and people tracking system.

➤ Vehicle tracking system

Mohammad A. Al-Khedher[9] from Al-Balqa University had done a hybrid GPS-GSM Localization of Automobile Tracking System. An integrated GPS-GSM embedded system had been used to track the current location of the vehicle by using Google Earth. The GPS receiver (MediaTek MT3329), microcontroller and GSM modem (SIM 900D) were mounted in the vehicle. Figure 2.6 shows the architecture of this system. The GPS receiver was used to receive the signal of the satellite. The microcontroller was used to read the certain engine parameters and transmit desired data to the server through the GSM modem. The second modem which is connected to the PC will receive the SMS that includes the GPS coordinate and engine parameter. Visual Basic program was used to convert the received SMS text to numerical form then saved in a Microsoft Office Excel file. In order to show the location of the automobile and the engine parameters on the map, the Microsoft Office Excel file was converted to KML (Keyhole Markup Language) format and Google Earth will interpret the KML file. This objective of this system is to manage a fleet, police automobiles distribution and car theft cautions.

Besides that, Kurnal Maurya[10] et al. also constructed a real time vehicle tracking system by using the same technology. A real time vehicle tracking system using GSM and GPS technology as an anti theft tracking system had been built for monitoring a moving vehicle continuously and report to the owner regarding the vehicle status.

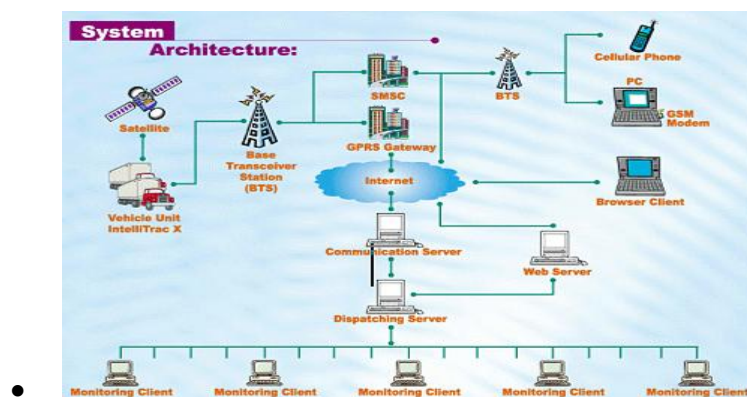


Figure2.6: The System architecture of vehicle tracking system

➤ Agricultural Monitoring System

Besides people and object tracking, GPS and GSM technology also can be used for agriculture monitoring. This research had been done by two of UTM undergraduate students, Gan Teck

Yik[11] and Arsany Bin Arsad[12] . They used the BSmart A300 b'AT Unit (concurrent GPS and GSM real time positioning technology) to monitor their plant, and receive warning messages during critical situations.

➤ **People Tracking System**

GPS and cellular network based system are commonly used to provide navigation services for outdoor environment. A UTM student, Aliff Fawzuan Othman had done a research based on the Hajj Pilgrim Tracking system GPS and GSM technology [13]. The tabung Haji staff will use the Android smart phone to request

the position of Hajj pilgrims while the Hajj pilgrims carried the GPS tracker or Smart phone. This system was using both GPS and GSM network technology for people tracking purpose.

➤ **People Tracking System**

This subtopic will discuss about the technologies that had been used for real time people positioning system and comparison among these technologies.

● **Wi-Fi(Wireless-Fidelity)**

Wi-Fi is a wireless network technology that can provide the wireless internet and network connection through radio wave. Amalina Abdul Halim from Mara University of Technology had done are search based on Wi-Fi positioning system[14]. It determined the user's location based on the Wi-Fi signal strength. By using the Wi-Fi, It is possible to determine the exact location of a person with accuracy up to 1 meter. The research used the real time location system from Ekahau to build their application, where Ekahau real time location system is a wireless radio frequency solution that is based on the Wi-Fi access point. People tracking system by using Wi-Fi technology is more economical because most of the communication devices are equipped with the wireless network. With a Wi-Fi receiver, the communication device can be used to determine a location. However, WI-Fi only can determine the user location at areas that have wireless reception. So It is not suitable to use in the rural area and this technology also needs an extensive mapping and high power consumption.

- **RFID(Radio Frequency Identification)**

The tag based system also can be used for tracking purposes. Tags based system includes RFID, barcode and Bluetooth. RFID, which stands for Radio Frequency Identification is an electronic device that belongs to a group of technology named Automatic Identification and Data Capture (AIDC). This technology can automatically identify the object and directly send the collected data to a computer.

Nezar Amer Alnizari [15] had done a research on real time people tracking by using this technology. For RFID technology, it can transfer the data from a tag which is attached to an object through the radio frequency. RFID consists of 3 important components, which are RFID tags, RFID reader and RFID PC reader. The RFID tag is used to send the RF signal to RFID reader, RFID reader is acting as a transceiver, receive the RF signal from the RFID and send the signal to the RFID PC reader. The received signals can be displayed on the PC through the RFID PC reader. People tracking system by using tag based technology is low cost compared to the other technology, but it only can be used locally and works only in the vicinity of tags.

- **A-GPS**

The Assisted Global Positioning System is another technology which can be used for positioning purposes. Albert Kai-Sun Wong and his partners [16] from Hong Kong University had designed an experimental AGPS-based elderly tracking system. It was a system which includes a wearable AGPS terminal, GPS Reference Station, location server, web client and web engine. In this research, the assistance data was retrieved by the wearable AGPS terminal by using the SUPL (Secure User Plane Location) protocol. For this system, AGPS terminal comprises of 3 main components, which are the 3G baseband chip (provides the terminal's HSPA (High Speed Packet Access) connectivity with the network, RF chips and GPS RF Chips (perform GPS positioning computation).

A device which is embedded with GPS receiver and ARM7 processor was used to receive the navigation message from the satellite and sent the signal to the AGPS controller and AGPS data store. AGPS is a technology which has faster time-to-first-fix and ability to provide an accurate

indication of one's location in outdoor and even the urban canyon environment. However, A-GPS cannot contact directly to the satellite. It needs to use other resources such as mobile network in order to contact to satellite. Thus, the user needs to pay for the mobile subscription fee.

Chapter 3

DESIGN OF THE TRACKING DEVICE

As we have discussed previously our goal is to design and analysis of a tracking device. We can use it in various applications. We are making it cost effective. This chapter will discuss about the approach and process flows used to complete this project. We will use show the devices work using block diagrams.

3.1 System Overview

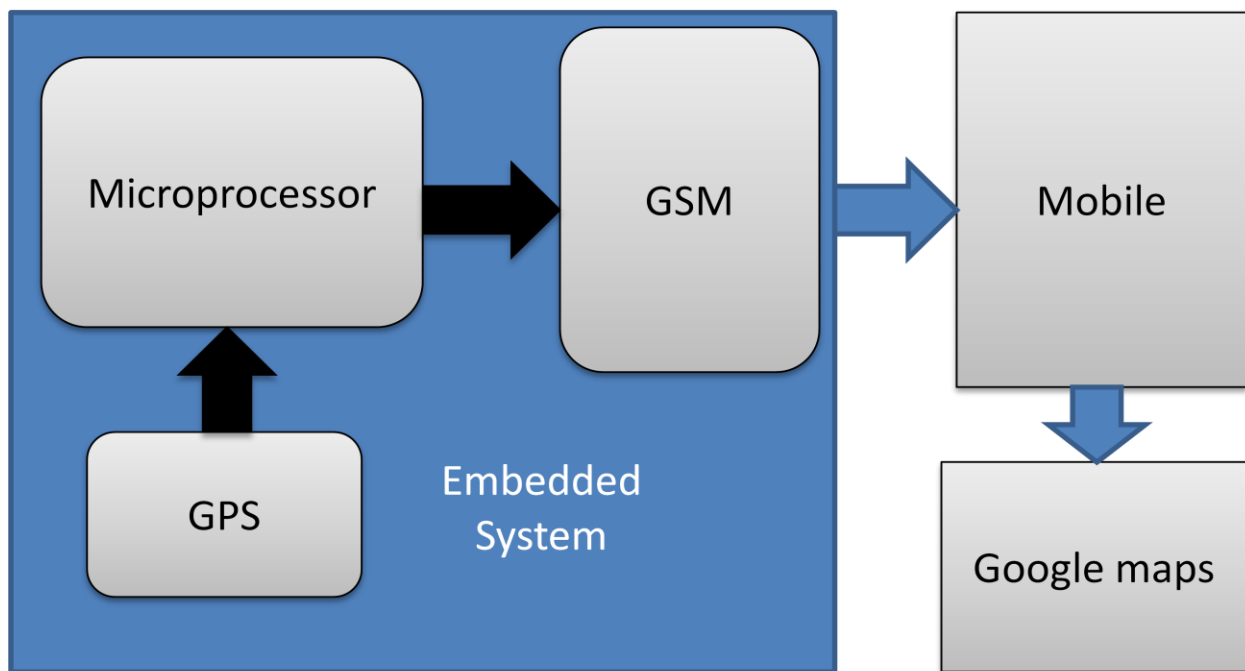


Fig3.1: Block Diagram of a tracking device

The GPS system receives signals from the satellites. It processes the data and sends it to the Microprocessor by using GSM it sends the data to the user mobile phone through an sms. The message contains the latitude, longitude and the altitude of the location. We can insert the latitude and longitude in the Google maps and get the exact location.

3.2 Design

Hardware Devices

The hardware devices used in this project are:

- Microcontroller(arduino uno)
- Transmitter and Receiver(GPS/GSM module)
- Power Source
- Antenna
- Cables and wirings

- **Microcontroller(Arduino UNO)**

Arduino UNO R3 is the latest version of the Arduino UNO board using the ATmega8U2 instead of the FTDI chip for faster transfer rate. It is a small form microcontroller board based on ATmega328. It is equipped with a USB connection, 32kBytes flash memory, 1 reset button, 14 digital input/output pins, 6 analog input pins and 1 power jack. The recommended input voltage for this board is between 7 to 12 volts. Thus, an external power supply is needed. Power can supply to this board through the power jack.

Besides that, Arduino UNO can communicate with a host PC by using a USB Type B cable. The user can write the program on the host PC and upload it to the board. After uploading the program, the USB Type B cable can be removed. The program will be stored in the Arduino Board and it will still run each time the reset button is pushed. Figure 3.2 shows the Arduino UNO Board.



Figure3.2: Microcontroller(Arduino UNO)

- **GPS/GSM Module**

GPS/GSM Module is a shield. It uses an embedded SIM908 chip and has Quad-band GPRS/GSM engine which works on the frequencies of 900 MHz, 1800 MHz, 850 MHz and 1900 MHz. It can be controlled using AT commands.

GPS/GSM Module is compatible with an Arduino Board such as Arduino UNO and Arduino Mega2560. Since this shield can support both of the GPS and GSM technology. It is suitable to use for tracking purpose at anywhere and anytime with signal coverage. Figure 3.3 shows the GPS/GSM Module.

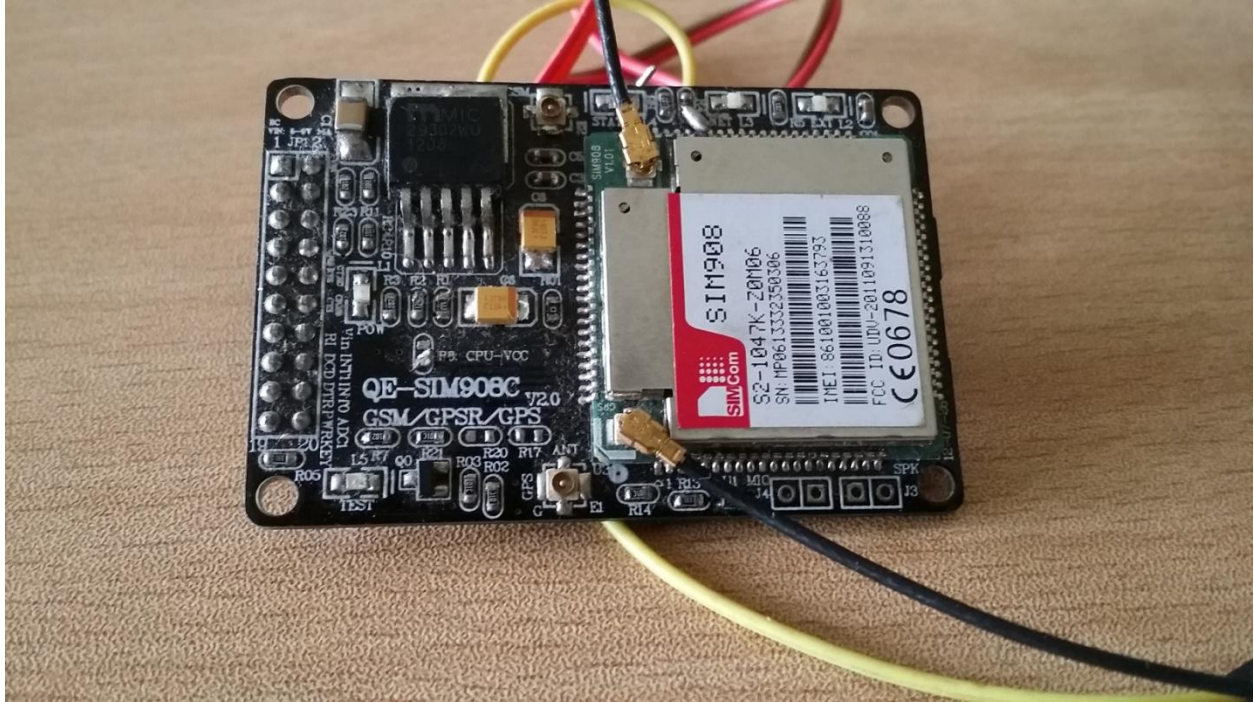


Figure3.3: GPS/GSM Module

- **Power Source:**

In our case used a battery as the power source for powering up the arduino and the module.

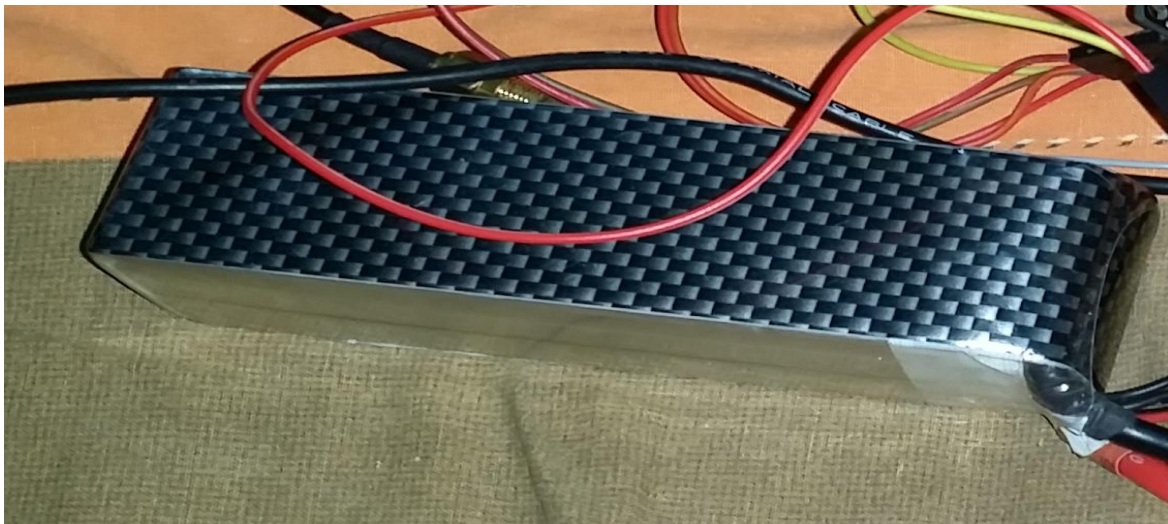


Figure3.4: Battery

- **Antenna:**

We used an antenna for the GSM communication. The Antenna for the GPS communication is installed with the module.



Figure3.5: Antenna

- **Cables and Wiring:**

Some cables and wires were used to make the connections among the devices.

There are two parts to this system. One is the tracking part and the other is the displaying part.

3.3 Development of a tracking Device:

The Tx pin of the arduino board is connected to the Rx pin of the gps/gsm module. And the Rx pin of arduino Tx pin of the gps/gsm module. The arduino board is powerd by a usb cable connected to a laptop. The gps/gsm module is powered by the battery. The antenna is connected to the gps/gsm module for sending and receiving the gsm data.

The figure shows the connections of all the devices in the system.

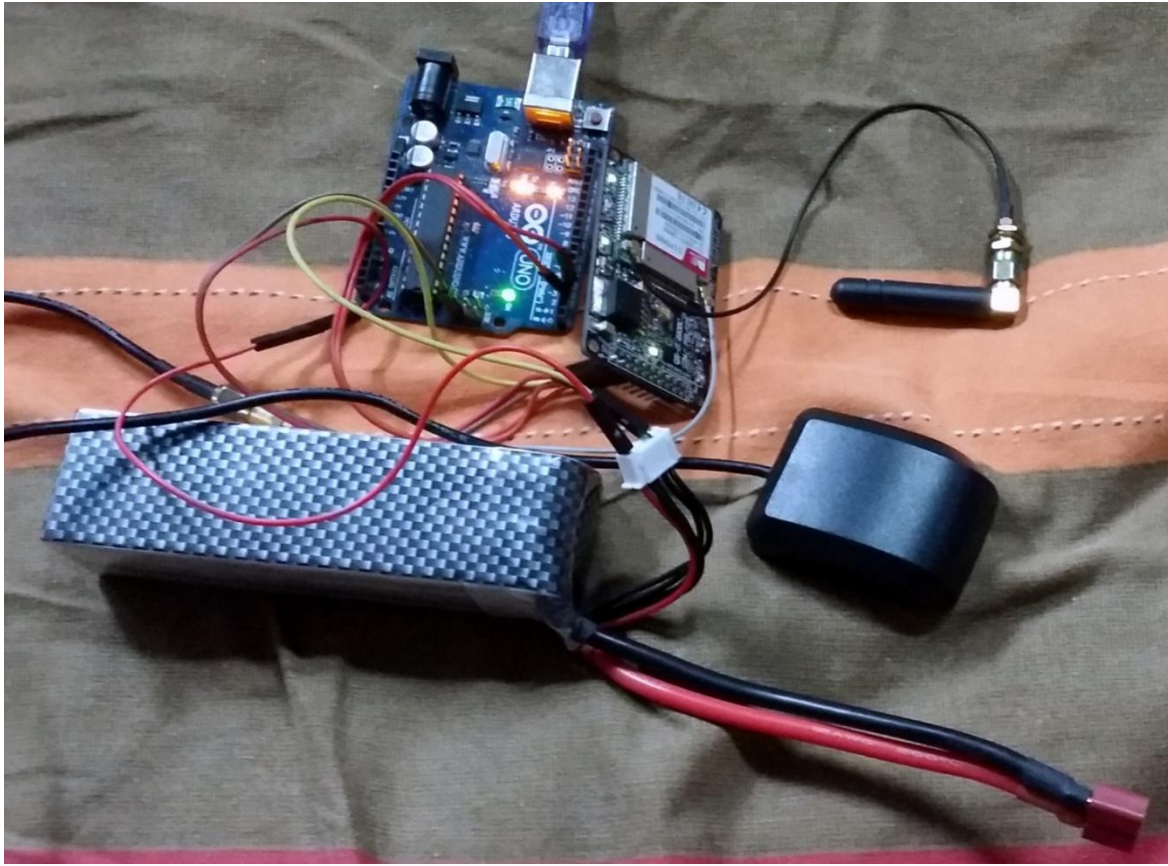


Fig 3.2: System Connection

After the whole connection is done the battery is connected and the GPS/GSM module turns on. Then the program is burned into the microcontroller. The GPS receives the positioning data. And the GSM transmits the data to the cellular device which in this case is a mobile phone. The phone number which was used was 01676982545. It was programmed in the microcontroller. Other numbers can also be added via the program. Then it will send the data through SMS to all those numbers.

3.4 Coding

The coding sequence can be shown in a block diagram as follows

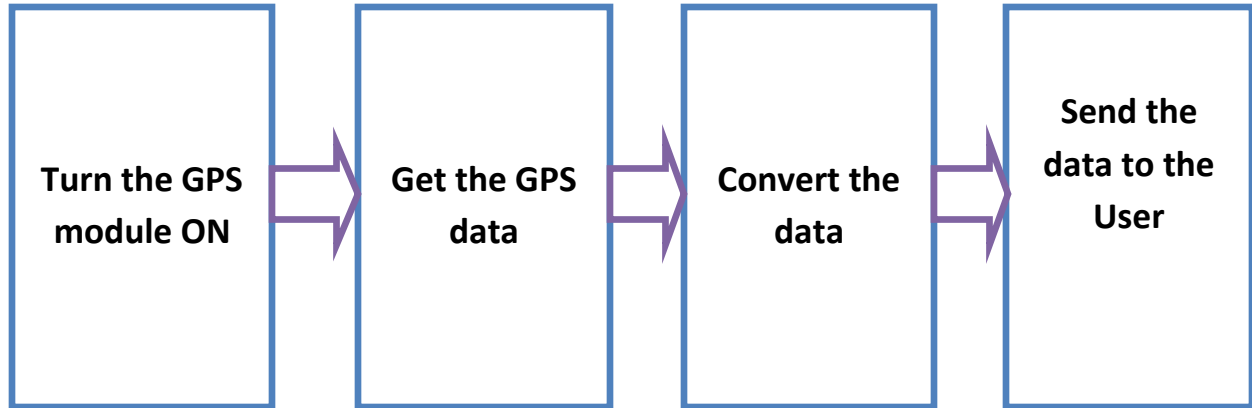


Fig3.3: Sequence of coding

At first, we turn the GPS module on by using the microcontroller. Then we receive the data from the satellites by the microcontroller. The data is received in the form of a string. Then the data is converted to degree, minutes and seconds, which is the real form for co-ordinates. Then the co-ordinates are sent to the user by the GSM module.

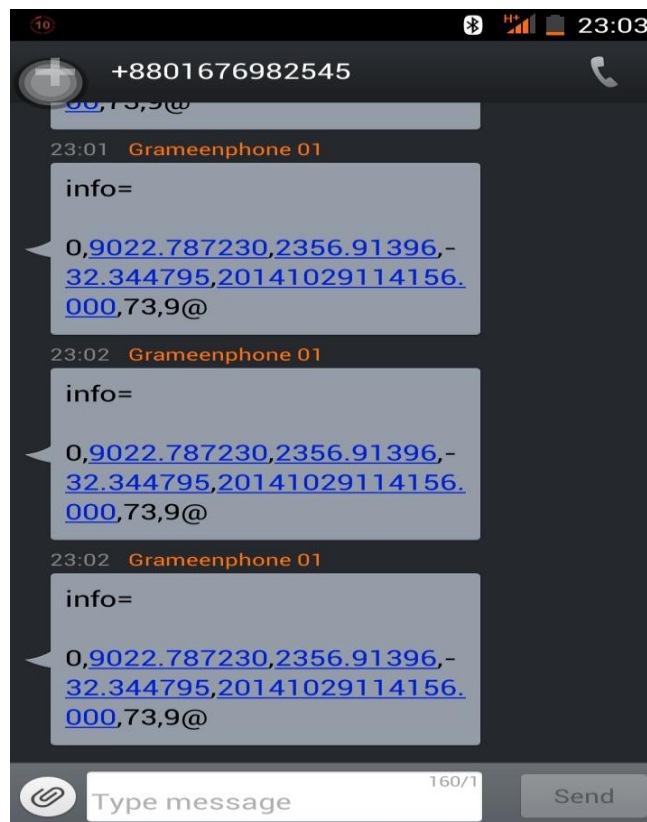
CHAPTER-4 TRACKING DATA ANALYSIS

After completing the setup, the entire system is tested to ensure that the people tracking system is working well and meets the requirement. This chapter will discuss about the results that are obtained from the testing the functionality of the system along with a cost comparison overview.

4.1 Tracking Results

At first we got signals from 2 satellites. But there was no data. According to the GPS theory we need at signals from at least 3 satellites to get the desired location. But in actual case we found out that we only got the location after we received signals from 9 satellites. At one stage we were getting signals from 15 satellites.

The results were sent to our mobile phone through sms. It looked like this:



The data we have in the sms is elaborated in the following table:

Parameter	Data received
Longitude	90 22.787230
Latitude	23 56. 91396
Altitude	-32.344795
Date and time	20141029114156(GMT)
Number of satellites	9
Velocity	0.000000

To get our desired location we need to put the latitude and longitude value into the Google maps and the map will show us the location.

Our main objective was to get a valid tracking data on our cellular device by using the GPS/GSM module along with the microcontroller. So we got our desired data on our cellular device by SMS. This is shown in the figure below.

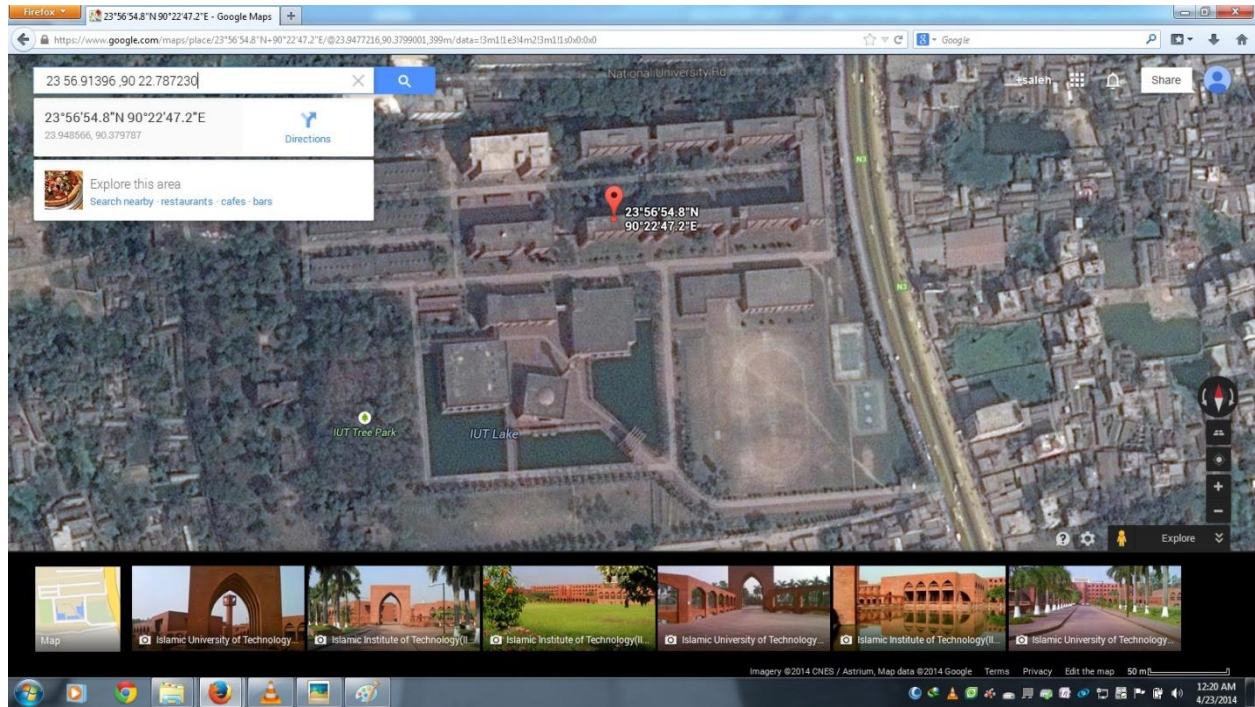


Fig 4.1: Matched results with Google Earth

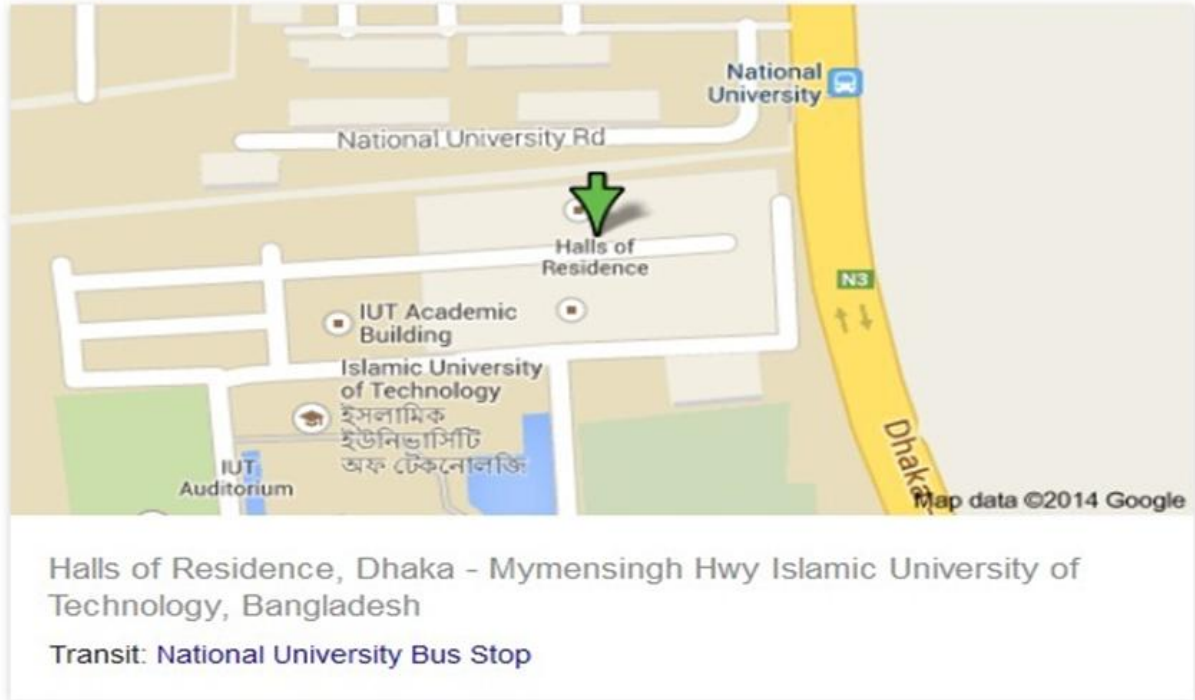


Fig 4.2: Matched results with google maps

4.2 Cost Analysis:

The main objective of our project is to make the tracker available for the general people of Bangladesh at a low cost.

- **Present cost:**

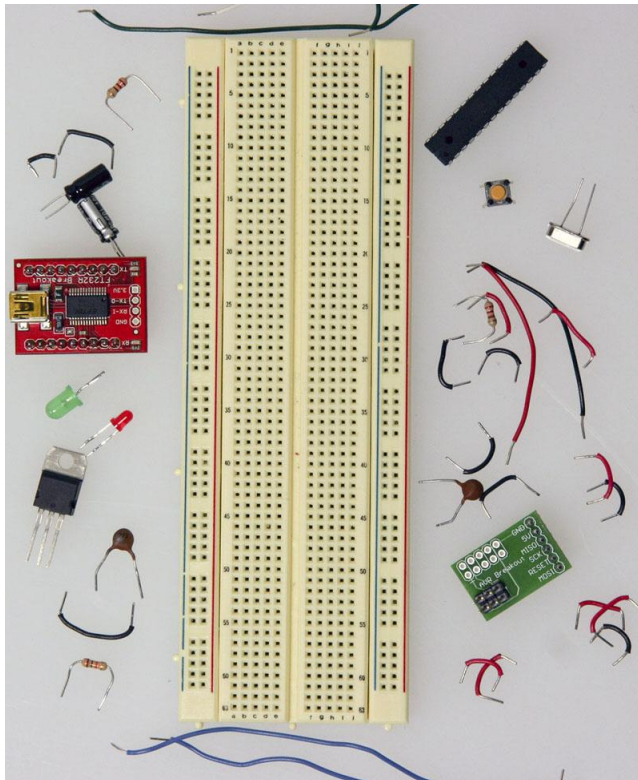
The cost of our Tracking Device is given below:

Equipments Used	Price(BDT)
Microprocessor(Arduino UNO)	800
GPS/GSM Module	3500
Antenna and Wiring	300
Battery	1000
Total	5600

- ⊙ The market value of a GPS Tracker in Bangladesh is around 8000 BDT.
- ⊙ So we have been able to reduce the cost somewhat.
- ⊙ But in our opinion it is still not affordable for the general people.
- ⊙ The price of a GPS tracker in the developed countries is around 5000 BDT.
- ⊙ We intend to lower the price even more.

Parts:

- A breadboard
- 22 AWG wire
- 7805 Voltage regulator
- 2 LEDs
- 2 220 Ohm resistors
- 1 10k Ohm resistor
- 2 10 uF capacitors
- 16 MHz clock crystal
- 2 22 pF capacitors



USB to Serial Communication Board:

We will need a FT232 USB Breakout board from SparkFun.

There are two options available from them:

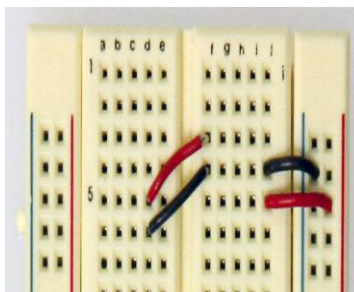
- FT232RL USB to Serial Breakout Board, SKU BOB-0071
- Arduino Serial USB Board, SKU DEV-08165

If we plan to use the top option and have not yet soldered headers to the breakout board, now would be a good time.

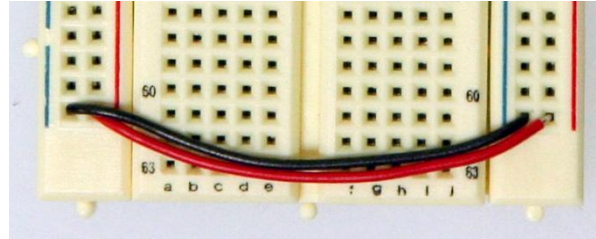
Bootloading the Atmega Chips:

There are several options for bootloading the Atmega chips, a few of which are covered in this tutorial. If we wish to bootload our Atmega chips our breadboard, an additional part will make our life much easier but is not necessary.

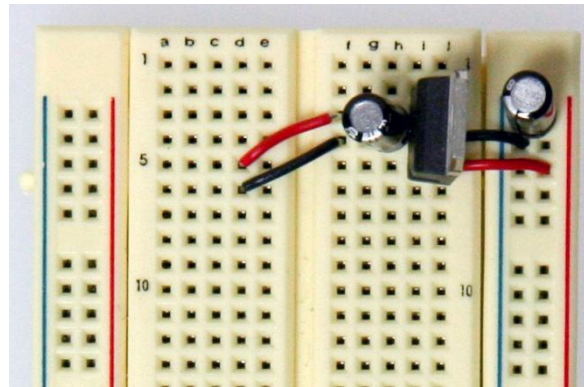
Adding circuitry for a power supply:



Power and ground wires for where our voltage regulator will be.

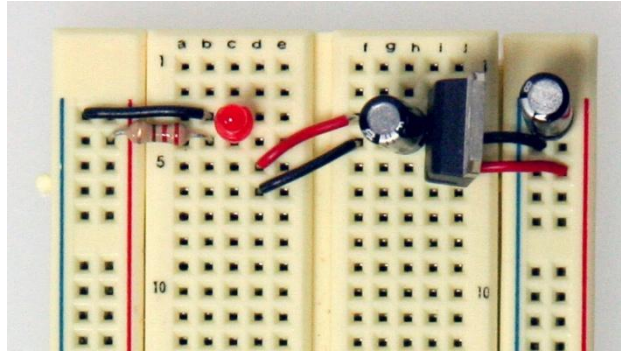


Power and ground wires at the bottom of our board connecting each rail.

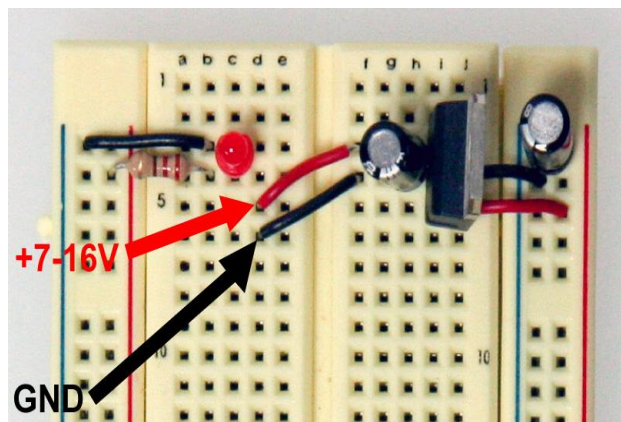


The 7805 power regulator and the lines to power the board. The regulator is a TO-220 package where the Input from the external power supply goes input on the left, ground is in the middle and the 5V output is on the right (when facing the front of the regulator). We need to add Power OUT and ground wires that connect to the right and left rails of the breadboard.

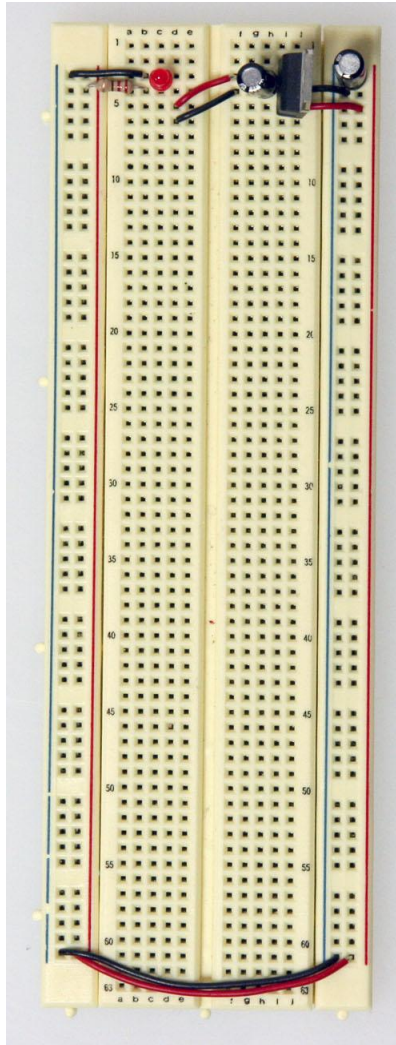
Also, We need to add a 10uF capacitor between the IN of the regulator and the ground as well as a 10uF capacitor on the right rail between power and ground. The silver strip on the capacitor signifies the ground leg.



An LED and a 220-ohm resistor on the left side of our board across from the voltage regulator. An LED attached to power like this is a great troubleshooting trick. We'll always know when our board is being powered as well as quickly know if our board is being shorted.



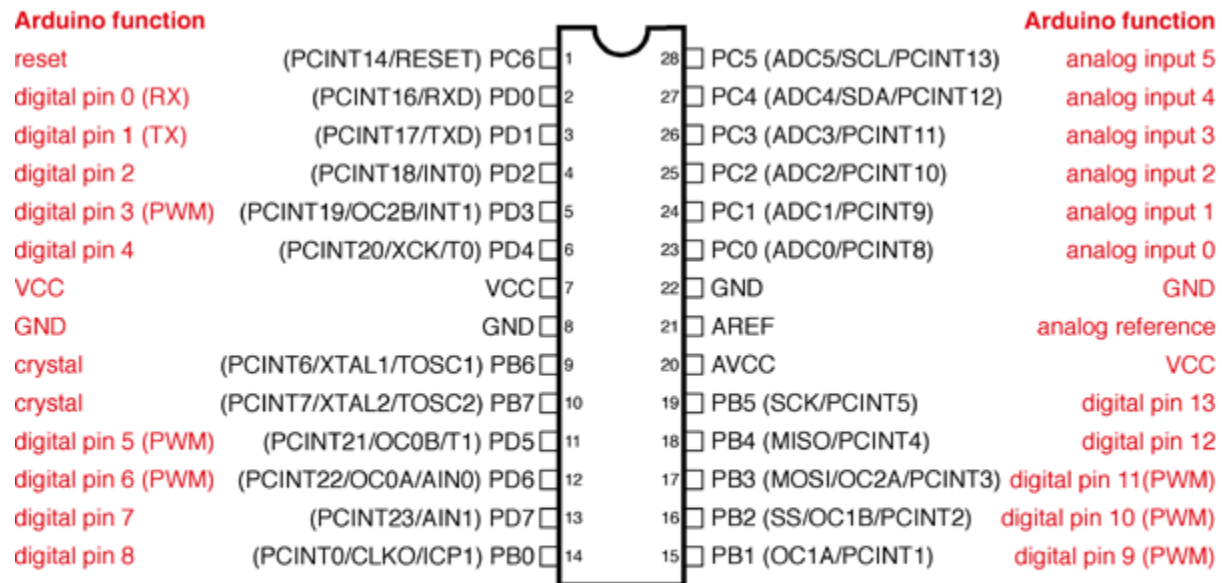
The red and black wires to the left of the voltage regulator is where our power supply will be plugged in. The red wire is for the POWER and the black wire is for the GROUND. We need to be sure to only attach a power supply that is between 7-16V. Any lower and we won't get 5V out of our regulator. Any higher and our regulator may be damaged. A 9V battery, 9V DC power supply, or 12V DC power supply is suitable.



Now that the power-basics are done we are ready to load on the chip.

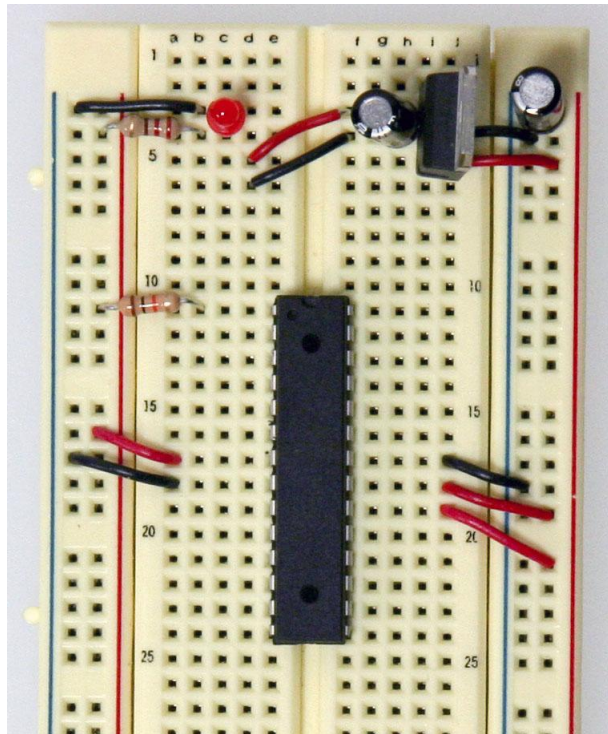
Microprocessor(ATMEGA8/168/328) Basics:

Atmega168 Pin Mapping



Digital Pins 11, 12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17, 18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

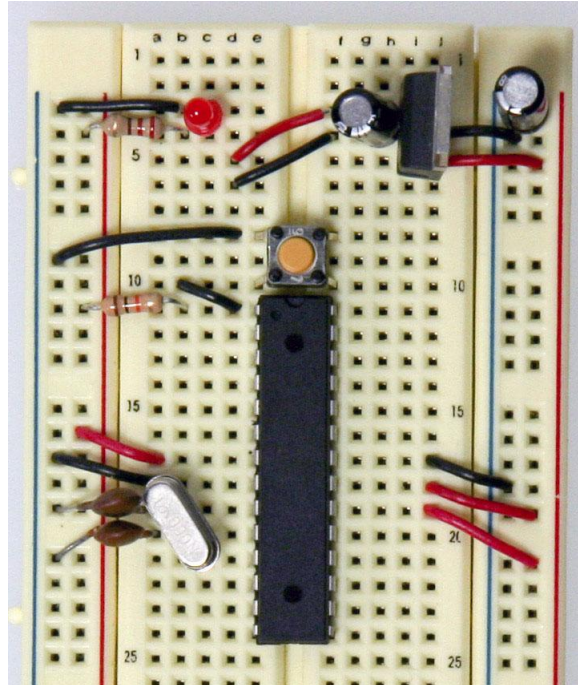
Before moving on, we can check out this image. It's a great resource for learning what each of the pins on our Atmega chip do in relation to the Arduino's functions. This will clarify a lot of confusion. For even more detailed information, we can take a peek at the datasheet for the Atmega 168.



Let's Start by connecting a 10k ohm pullup resistor to +5V from the RESET pin in order to prevent the chip from resetting itself during normal operation. The RESET pin reboots the chip when pulled down to ground. In later steps we will show how to add a reset switch that takes advantage of this.

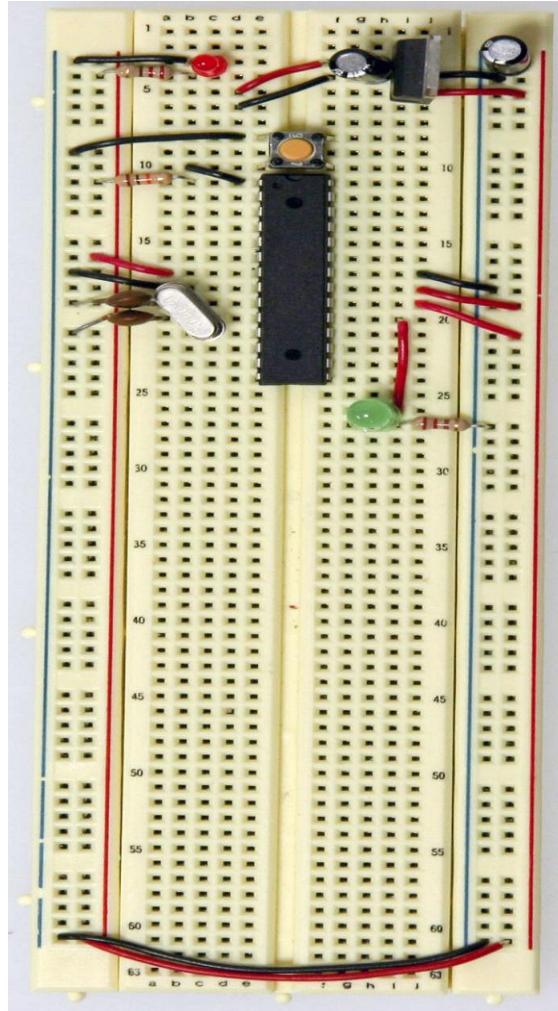
- Pin 7 - Vcc - Digital Supply Voltage
- Pin 8 - GND
- Pin 22 - GND
- Pin 21 - AREF - Analog reference pin for ADC
- Pin 20 - AVcc - Supply voltage for the ADC converter. Needs to be connected to power if ADC isn't being used and to power via a low-pass filter if it is (a low pass filter is a circuit that reduces noise from the power source. This example isn't using one)

the chip when needed. We add the switch just above the top of the Atmega chip crossing the gap in the breadboard. Then, add a wire from the bottom left leg of the switch to the RESET pin of the Atmega chip and a wire from the top left leg of the switch to ground.



The chip used on this board is actually already programmed using the `blink_led` program that comes with the Arduino software. If you already have an Arduino printed circuit board running, it is a good idea to go ahead and check the breadboard version you are building with a chip we know works. We pull the chip from our working Arduino and try it on this board. The `blink_led` program blinks pin 13. Pin 13 on the Arduino is NOT the AVR ATMEGA8-16PU/ATMEGA168-16PU pin 13. It is actually pin 19 on the Atmega chip.

We Refer to the [pin mapping](#) above.



At this point if we had already programmed our chip somewhere else and didn't need this breadboard circuit to reprogram the chip, we could stop here. But now we will see about the boot loading of our chip on the breadboard.

Burning the Microchip:

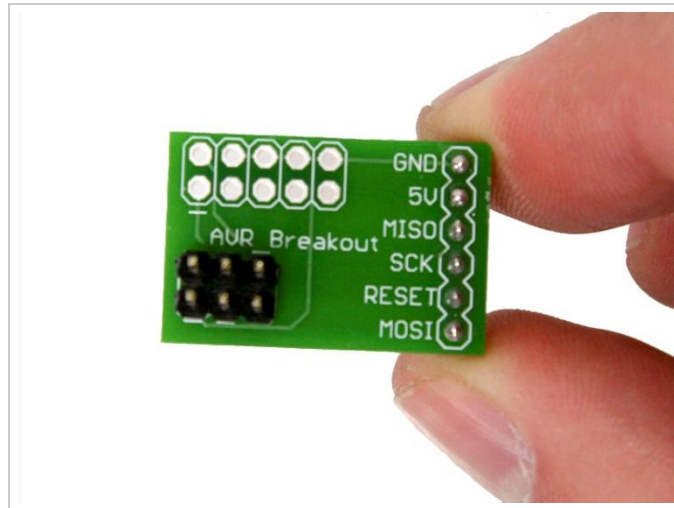
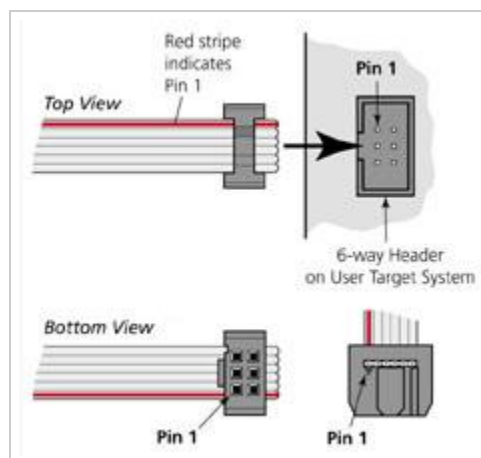


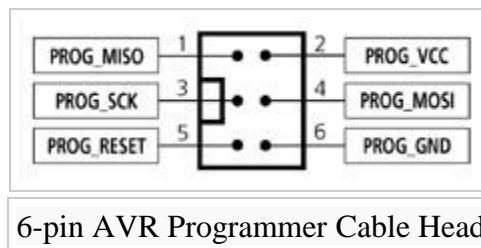
Fig:AVR Programming Adapter

When bootloading an Atmega chip on a breadboard, the AVR programming adapter (SKU [BOB-08508](#)) from Sparkfun is incredibly handy. This adapter breaks out the 6 pins from the programmer to 6 inline pins for easy attachment to the breadboard. All the pins are also labeled making it very easy to connect it up to our chip.

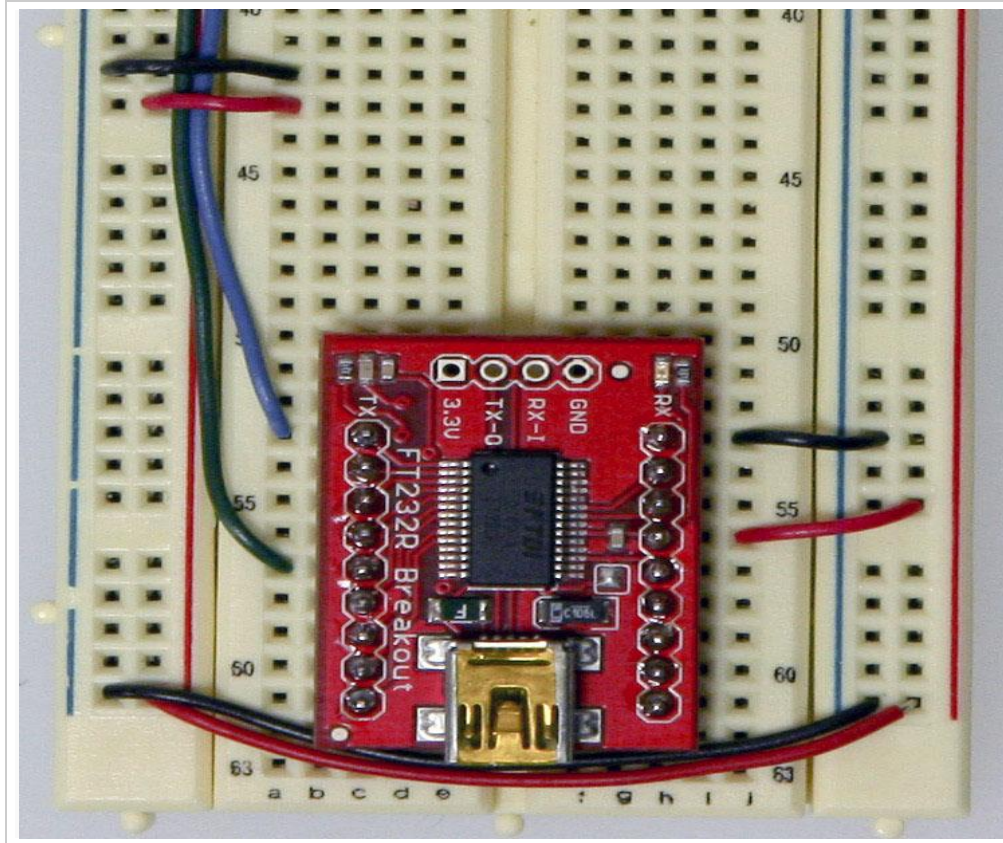


6-pin AVR Programmer Cable

If we don't have an AVR programming adapter we can still bootload without it. It will however be more of a headache to set up. The two images to the left are great references when hooking up a programmer to an Atmega chip without an adapter board. The images will tell us what all the holes in the 6-pin AVR plug are and we will simply need to stick wires in the end and run them to our Atmega chip.

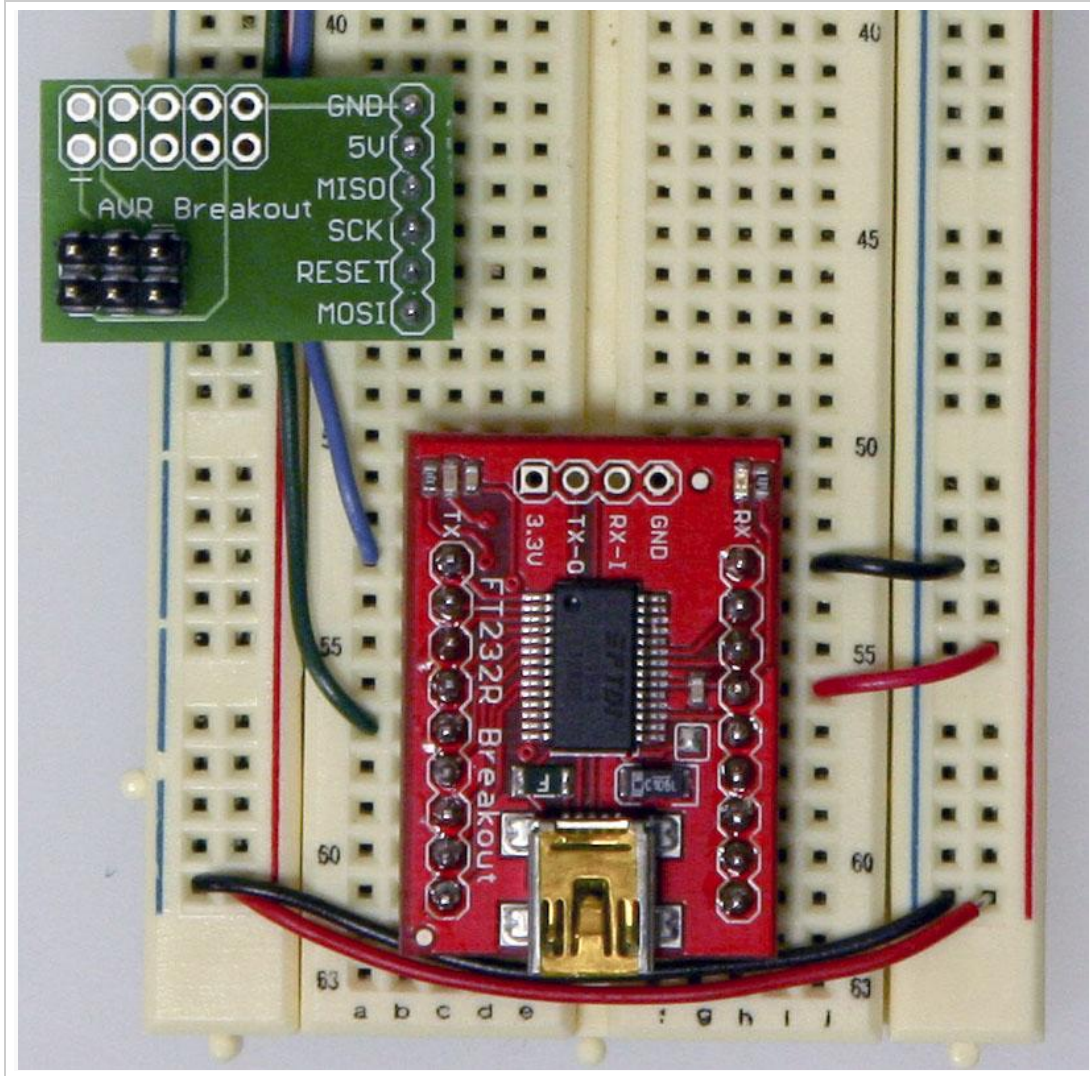


This image is a view from the bottom and labels each of the holes. Take note of the square as to what orientation our cable is I



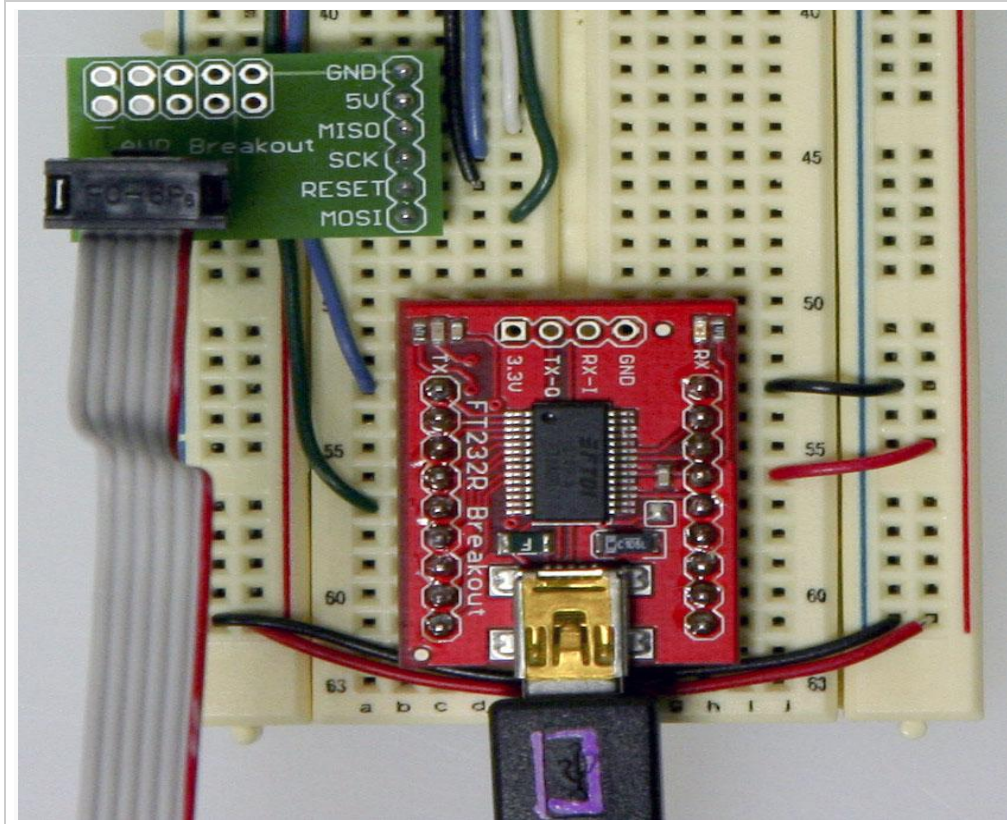
Adding power and ground

With the breadboard we prepared above, add two wires for power and ground for our AVR programmer.



Plugging in the AVR adapter

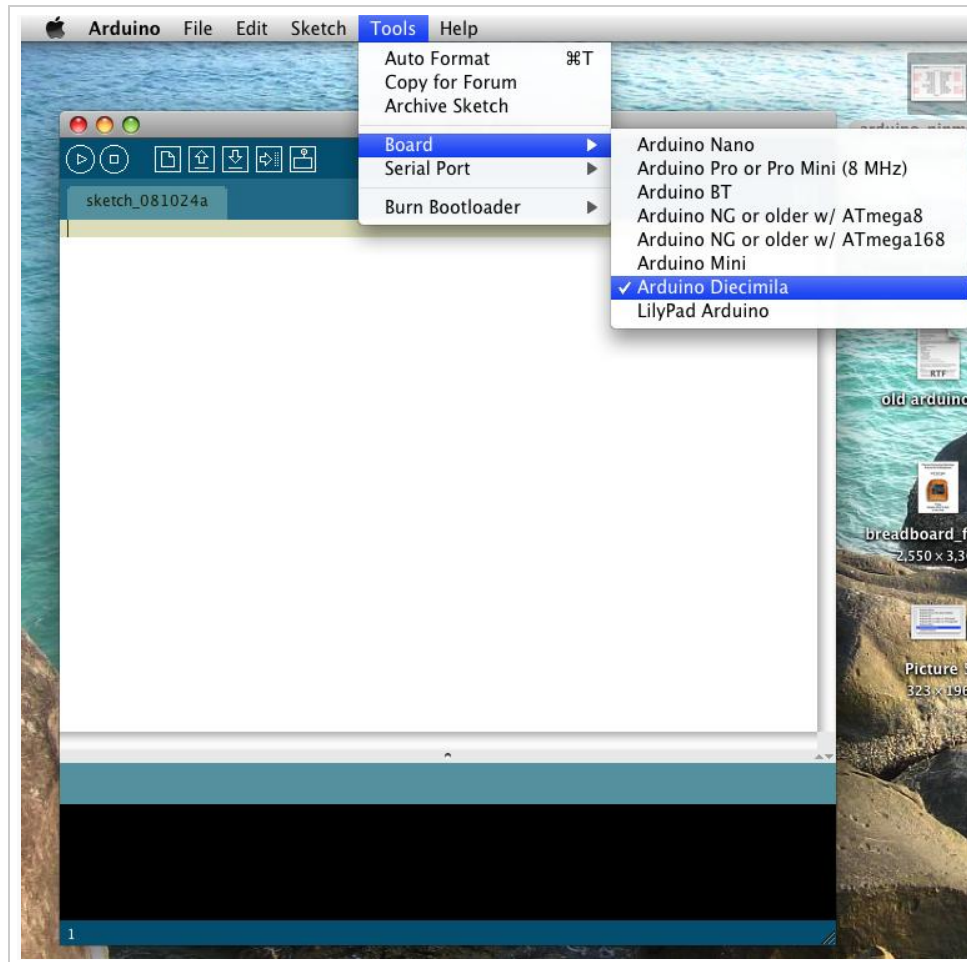
Now plug the AVR programming adapter into the breadboard with the GND pin matching up with the ground wire we just ran and the 5V pin matching up with the power wire we just ran.



Plugging in the USB cable and AVR programming cable

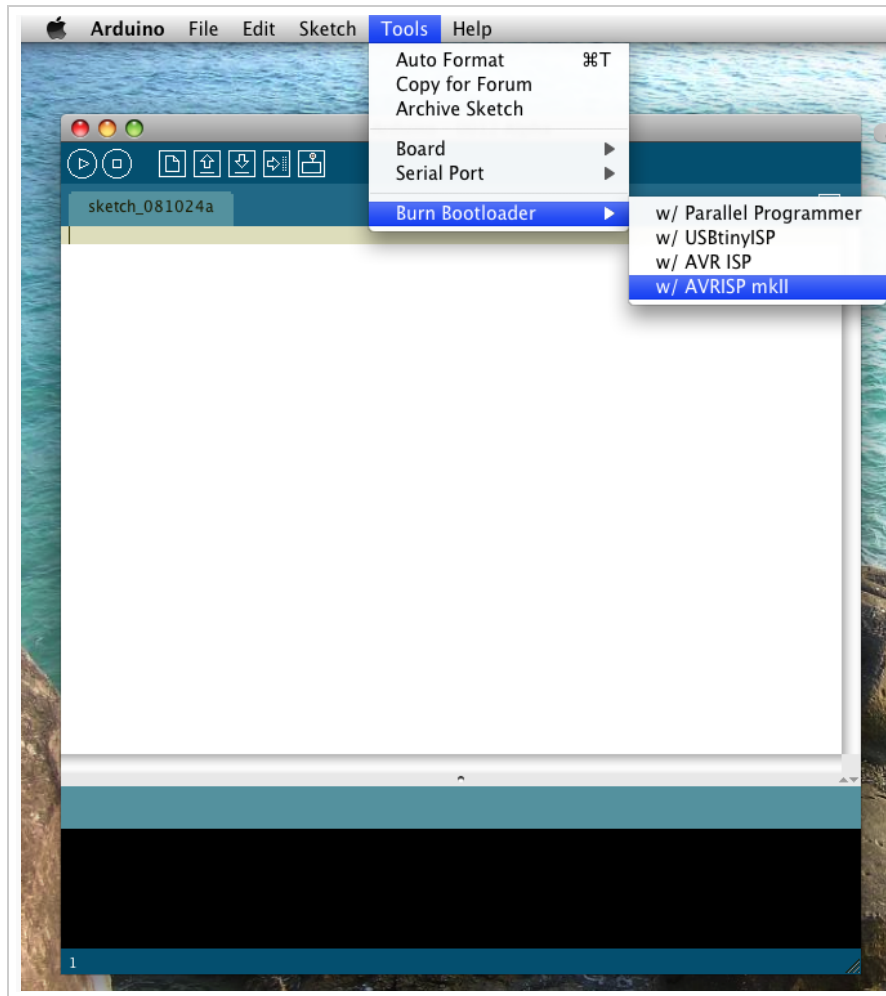
We are almost there! Just plug in a USB cable to our USB breakout board and plug the 6-pin plug of our AVR programmer to your AVR programming adapter. The black nub of the 6-pin head must be facing upwards towards the Atmega chip.

In the next step, we'll show you how to use the Arduino software to burn your bootloader!



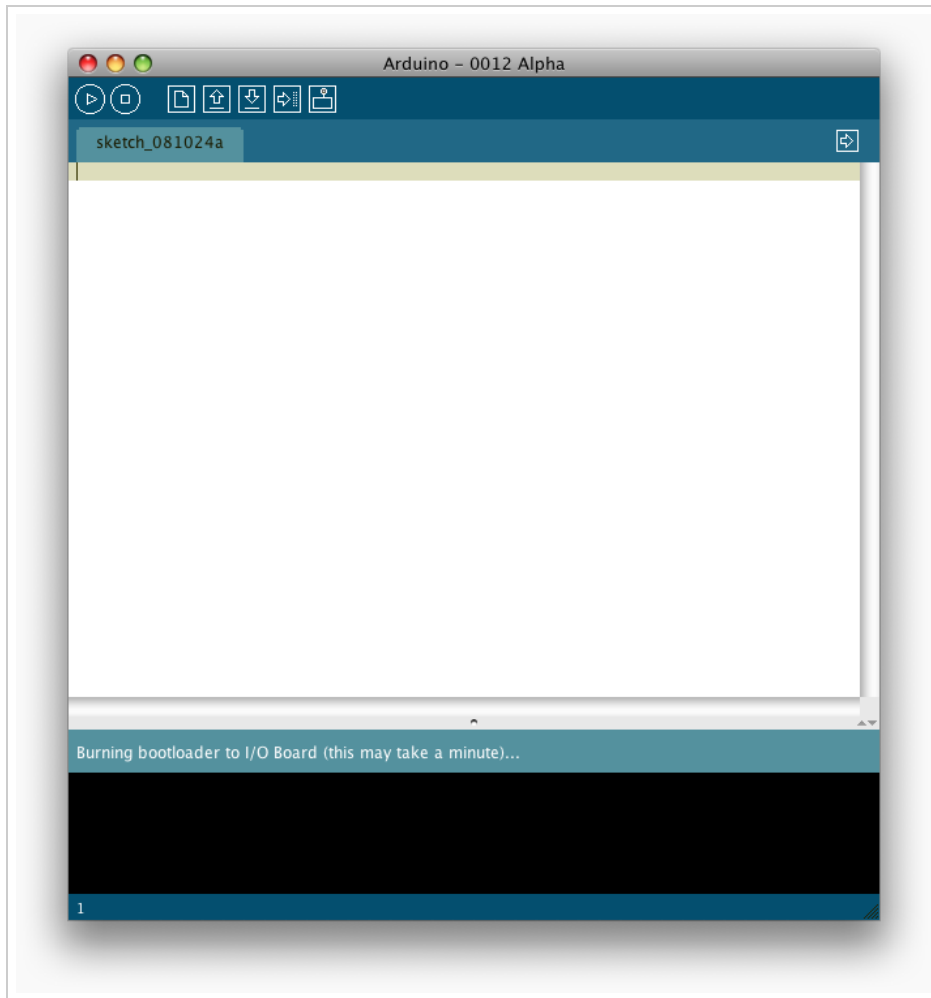
Picking our board type

We Fire up Arduino and then go to 'Tools' and 'Board'. Choosing the type of board we'd like to use will effect which bootloader we will be put on your chip. Most commonly we will be using the Diecimilia or the most recent version of Arduino for an Atmega PDIP, however if you'd like to bootload an Arduino Lilypad, Arduino Mini, Arduino Nano, or any of the older Arduino versions, choose the appropriate board.



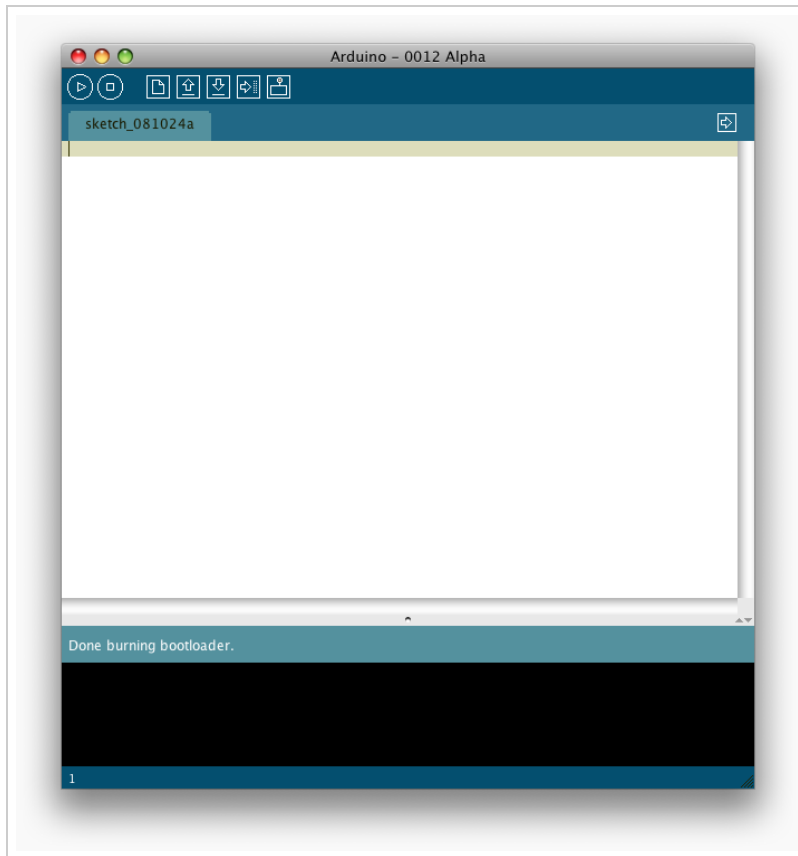
Choosing our programmer

Then, we go to 'Tools' and 'Burn Bootloader' and choose the programmer we will be using.



Burning

Once we chose our programmer, the AVR programmer will begin bootloading our Atmega chip and a message will appear in the status bar which reads "Burning bootloader to I/O Board (this may take a minute)..." Lights will flicker on our programmer.



Burn Complete

When done bootloading, the status bar will be updated with the message "Done burning bootloader." Our chip is now ready to be programmed using the Arduino software! Congrats! Power cycle our Arduino and our new Atmega chip will be running a simple LED blink program with pin 13 (if this is not the case, try programming it with one). If this is working, it was most definitely a success.

If we use the microchip of the microprocessor on the breadboard we can reduce the cost considerably. In that case the cost becomes:

Parts	Price
A breadboard	100
22 AWG wire	10
7805 Voltage regulator	20
2 LEDs	10
2 220 Ohm resistors	20
1 10k Ohm resistor	30
2 10 uF capacitors	60
16 MHz clock crystal	50
2 22 pF capacitors	80
Total	380

So the price of our tracking device becomes 5180BDT. By implementing the microchip in the breadboard we can reduce the price of our current tracking device even more.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

Our main goal was to minimize the costs associated in designing an efficient tracking device. We have somewhat succeeded in our task. We have reduced the cost by a notable amount. Then again we also wanted to show that this cost efficient design can work precisely. This, We have done by matching our data with the google maps and google earth. Now we would like to recommend about further edits to our project. We

5.2 Recommendations

1. Real time tracking can be introduced to our device by using a server and an interfacing.
2. Implementation of our device in a practical scenario is also recommended.
3. The use of assisted GPS is also recommended.

REFERENCES

1. El-Rabbany, A. Introduction to GPS: The Global Positioning System (2nd Edition). Canada: Artech House Publishers. 2006.
2. Broida, P. How to Do Everything with Your GPS. London. London: McGraw- Hill/Osborne. 2004.
3. Xu, G. GPS: Theory, Algorithms and Applications (2nd Edition). Germany: Springer. 2007.
4. NMEAData.URL<http://www.gpsinformation.org/dale/nmea.htm>.
5. Tisal, J. The Gsm Network: GPRS Evolution: One Step Towards Umts.(2nd ed.). England: John Wiley. 2001.
6. Hsiao, W. and Chang, S. The Optimal Location Update Strategy of Cellular Network Based Traffic Information System. Intelligent Transportation Systems Conference. 2006.
7. Fares, A. Network Switching Subsystem Engineering: The NSS Protocol architecture. German: Artech House Publishers. 2005.
9. Al-Khedher, M. A. Hybrid GPS-GSM Localization of Automobile Tracking System. International Journal of Computer Science And Information Technology (IJCSIT), 2011. 3(6): 75–85.
10. Maurya, K.,Singh, M.and Jain,N. Real Time Vehicle Tracking System using GSM and GPS Technology- An Anti-theft Tracking System. International Journal of Electronics and Computer Science Engineering. 1(3): 1103–1107.
11. Yik, G. T. Application of GPS and GSM Tracking Device In Agriculture Monitoring. Master's Thesis. Univetsiti Teknologi Malaysia,Skudai. 2009.
12. Arsad, A. B. Agriculture Monitoring System Powered By Global Positioning System(GPS) And GSM Wireless Technology. Master's Thesis. Univetsiti Teknologi Malaysia. 2009.
13. Othman,A.F.Development Of Hajj PilgrimTracking System.Master'sThesis. Univetsiti Teknologi Malaysia. 2009.

14. Halim, A. A. WIFI Positioning System. Master's Thesis. MARA Univetsiti of Teknologi Shah Alam. 2006.
15. Alnizari, N. A. A real-time tracking system using RFID in Mecca. Master's Thesis. Massey University. 2011.
16. sun Wong, A. K., Woo, T. K., Lee, A. T.-L., Xiao, X. and Luk, V. W.-H. An AGPS-Based Elderly Tracking System. Hong Kong University of Science and Technology Clearwater Bay, Hong Kong. 20. GPS/AGPS chip set targets telematics, navigation apps. Electronic Engineering Times, 2004.

Appendix

Code for turning On the GPS module

```
void start_GPS(){
  //mySerial.println("starting gps");
  unsigned long previous;
  previous = millis();// starts the GPS
  sendATcommand("AT+CGPSPWR=1", "OK", 2000);
  sendATcommand("AT+CGPSRST=0", "OK", 2000);
  while(( sendATcommand("AT+CGPSSTATUS?", "2D Fix", 5000) || //
waits for fix GPS
  sendATcommand("AT+CGPSSTATUS?", "3D Fix", 5000)) == 0 ) &&
  ((millis() - previous) < 20000))
  {
    if ((millis() - previous) < 20000)
      { Serial.println(F("fixed"));
        }
    else
      {
        Serial.println(F("not fixed"));
      }
  }
}

void closegps()
{
```

```

sendATcommand("AT+CGPSPWR=0", "OK", 2000);
}

int8_t check_fix(){
//check for power on
if((sendATcommand("AT+CGPSSTATUS?", "2D Fix", 5000)) == 1 )
{
    Serial.println(F("2d fix"));
    twoDfix=true;
    return 1;

}
else if((sendATcommand("AT+CGPSSTATUS?", "3D Fix", 5000)) == 1 )

{
    Serial.println(F("3D fix"));
    threeDfix=true;
    return 1;
}
else

{
    Serial.println(F("no fix"));
    twoDfix=false;
    threeDfix=false;
    return 0;

}

}
}

```

Code for receiving the GPS data in the microcontroller

```
int8_t get_GPS(){

    int8_t counter, answer;
    unsigned long previous;

    // First get the NMEA string
    // Clean the input buffer
    while( mySerial.available() > 0) mySerial.read();

    // request Basic string

    sendATcommand("AT+CGPSINF=0", "AT+CGPSINF=0\r\n\r\n", 2000);

    counter = 0;
    answer = 0;
    memset(frame, '\0', 100); // Initialize the string
    previous = millis();
    // this loop waits for the NMEA string
    do{

        if(mySerial.available() != 0){
            // Serial.print(mySerial.read());
            frame[counter] = mySerial.read();
            counter++;
            // check if the desired answer is in the response of the module
            if (strstr(frame, "OK") != NULL)
            {
                answer = 1;
            }
        }
    }
}
```

```

    }
}
// Waits for the answer with time out
}
while((answer == 0) && ((millis() - previous) < 2000));

frame[counter-3] = '\0';

// Parses the string
strtok(frame, ",");
strcpy(longitude, strtok(NULL, ",")); // Gets longitude
strcpy(latitude, strtok(NULL, ",")); // Gets latitude
strcpy(altitude, strtok(NULL, ".")); // Gets altitude
strtok(NULL, ",");
strcpy(date, strtok(NULL, ".")); // Gets date
strtok(NULL, ",");
strtok(NULL, ",");
strcpy(satellites, strtok(NULL, ",")); // Gets satellites
strcpy(speedOTG, strtok(NULL, ",")); // Gets speed over ground. Unit is knots.
strcpy(course, strtok(NULL, "\r")); // Gets course

convert2Degrees(latitude);
convert2Degrees(longitude);

// if (latitude[0]=='0' || longitude[0]=='0' || twoDfix==false || threeDfix==false)
// {
//   senddata=false;
//   mymySerial.println(F("gps zero"));
//   gpsfail++;
// }
// else if (!strcmp(latitude, prevlat) && !strcmp(longitude, prevlong))

```

```

// {
//  senddata=false;
//  mymySerial.println(F("gps same"));
//
// }
//
//else
//{
//  gpsfail=0;
//
//}

return answer;
}
void wait_time_sec(unsigned long int time)
{ unsigned long int timeDeal;
  timeDeal=millis();

  while((millis() - timeDeal) < (time*1000)){

  }

}
void wait_time_milli(unsigned long int time)
{ unsigned long int timeDeal_milli;
  timeDeal_milli=millis();
  while((millis() - timeDeal_milli) < time){
  }
}

```

Code for converting the GPS data

```
/* convert2Degrees ( input ) - performs the conversion from input
 * parameters in DD°MM.mmm' notation to DD.ddddd° notation.
 *
 * Sign '+' is set for positive latitudes/longitudes (North, East)
 * Sign '-' is set for negative latitudes/longitudes (South, West)
 *
 */
```

```
int8_t convert2Degrees(char* input){
```

```
    //Serial.println(F("converting"));
```

```
    float deg;
```

```
    float minutes;
```

```
    boolean neg = false;
```

```
    //auxiliar variable
```

```
    char aux[10];
```

```
    if (input[0] == '-')
```

```
    {
```

```
        neg = true;
```

```
        strcpy(aux, strtok(input+1, "."));
```

```
    }
```

```
    else
```

```
    {
```

```
        strcpy(aux, strtok(input, "."));
```

```
    }
```

```
    // convert string to integer and add it to final float variable
```

```
    deg = atof(aux);
```

```
strcpy(aux, strtok(NULL, '\0'));
minutes=atof(aux);
minutes/=1000000;
if (deg < 100)
{
    minutes += deg;
    deg = 0;
}
else
{
    minutes += int(deg) % 100;
    deg = int(deg) / 100;
}

// add minutes to degrees
deg=deg+minutes/60;

if (neg == true)
{
    deg*=-1.0;
}

neg = false;

if( deg < 0 ){
    neg = true;
    deg*=-1;
}

float numeroFloat=deg;
```



```

int parteEntera[10];
int cifra;
long numero=(long)numeroFloat;
int size=0;

while(1){
    size=size+1;
    cifra=numero% 10;
    numero=numero/10;
    parteEntera[size-1]=cifra;
    if (numero==0){
        break;
    }
}

int indice=0;
if( neg ){
    indice++;
    input[0]='-';
}
for (int i=size-1; i >= 0; i--)
{
    input[indice]=parteEntera[i]+'0';
    indice++;
}

input[indice]='.';
indice++;

numeroFloat=(numeroFloat-(int)numeroFloat);
for (int i=1; i<=6 ; i++)
{

```

```
numeroFloat=numeroFloat*10;
cifra= (long)numeroFloat;
numeroFloat=numeroFloat-cifra;
input[indice]=char(cifra)+48;
indice++;
}
input[indice]='\0';

}
```

Coding for Sending the converted data to the user data

```
int8_t sendATcommand(char* ATcommand, char* expected_answer1, unsigned int timeout){

uint8_t x=0, answer=0;
char response[200];
volatile unsigned long previous;

memset(response, '\0', 200); // Initialize the string

wait_time_milli(100);

while( mySerial.available() > 0) mySerial.read(); // Clean the input buffer

mySerial.println(ATcommand); // Send the AT command

x = 0;
previous = millis();

// this loop waits for the answer
do{
if(mySerial.available() != 0){
//Serial.print(mySerial.read());
response[x] = mySerial.read();
x++;
// check if the desired answer is in the response of the module
if (strstr(response, expected_answer1) != NULL)
{
answer = 1;
}
}
}
```

```
// Waits for the answer with time out
}
while((answer == 0) && ((millis() - previous) < timeout));

return answer;

}
```