**Security on Touch!**

Prototype of the Geolocation Tracker

**Mentor : Mr. Abhay Kshirsagar (Etrx dept)**

**Team Members: (Cmpn dept)**

**1. Kirti Jaisinghani**

**2. Jigar Dedhia**

**3. Aakash Goplani**

|  |  |  |
| --- | --- | --- |
| SR. NO. | CONTENTS | PG.NO. |
|  | INTRODUCTION | 1 |
|  | SURVEY | 2 |
|  | SYSTEM OVERVIEW | 3 |
|  | BLOCK DIAGRAM | 4 |
|  | HARWARE SPECIFICATION | 5-14 |
|  | SOFTWARE SPECIFICATION | 15-17 |
|  | APPLICATIONS | 18 |
|  | FUTURE SCOPE | 19 |
|  | REFERENCES | 19 |
|  | ANNEXURE | 20- |

**INTRODUCTION:**

The increasing rate of crime and life threatening attacks on human beings has aroused the need of a personalized security system. In order to reduce these increasing crimes, we need a system wherein the police can be informed about the person under attack and its location, so as to rescue them well in time. This project is an idea to make a device (prototype) which sends the location & identity of the person under attack to a number via a sms text with just a click of a button. If possible to design this device using nano-technology, this device can be embedded into any item that person carries on daily basis. If such a device is incorporated in the security system for the people, this world may be a safer place to live. To implement the prototype of this device a basic arduino board (freeduino) is interfaced with a GPS module & a GSM module.

* Ever felt the fear of getting lost in an unknown place?

Yes. This device helps u feel safe anywhere you go. Just with one click of a button your location is known to a person you trust.

* To design a personalized security wireless module; that can communicate with a mobile phone.

This would ensure:

* To provide basic sense of security to people who work late at nights.
* Availability of a security device to common man at an affordable price.
* To reduce the crime rate in our country.
* Explore the world without getting lost. Go to exotic places and explore the world whenever you feel like it.

What’s Arduino?

It’s an open source electronics prototyping system using popular proven tools and hardware. It’s designed to be easy to interface with sensors and actuators, and inexpensive enough you can use many in your projects!

Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino projects can be stand-alone or they can communicate with software running on a computer (e.g. Flash, Processing, MaxMSP).

The boards can be built by hand or purchased preassembled; the software can be downloaded for free. The hardware reference designs (CAD files) are available under an open-source license, you are free to adapt them to your needs.

Arduino received an Honorary Mention in the Digital Communities section of the 2006 Ars Electronica Prix. The Arduino founders are: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis. Credits

The Arduino programming language is based on a very easy-to-learn open-source language called Wiring, which is similar to C/C++, but streamlined for quicker development!

**Hardware specifics:**

An Atmel ATmega328 running at 16MHz, connected through an FTDI232R USB-to-serial converter (just plug it into a USB port, and go!). 13 digital I/O pins (6 with PWM) and 6 analog inputs. Power comes from your USB port, or external 6-20VDC power supply. Your PC is protected by a 500mA resettable fuse. LEDs for power, Rx/Tx activity and pin 13 “test” are all pre-installed!

**Survey:**

As a part of the project review we took a basic survey for college students asking them -- how safe do they feel in this city?

Majority of the students felt unsafe.

Few felt a little safe if they live near-by.

And a handful of students felt completely safe probably hostelites.

**System overview:**

**System**

**Hardware specification Software specification**

1. Block diagram. 1. Algorithm

2. Components description. 2. Platform used

3. Flow of the program

**Hardware specification:**

Block Diagram: General connections of the project, of all the components.

Component Description: Detailed description of the components used.

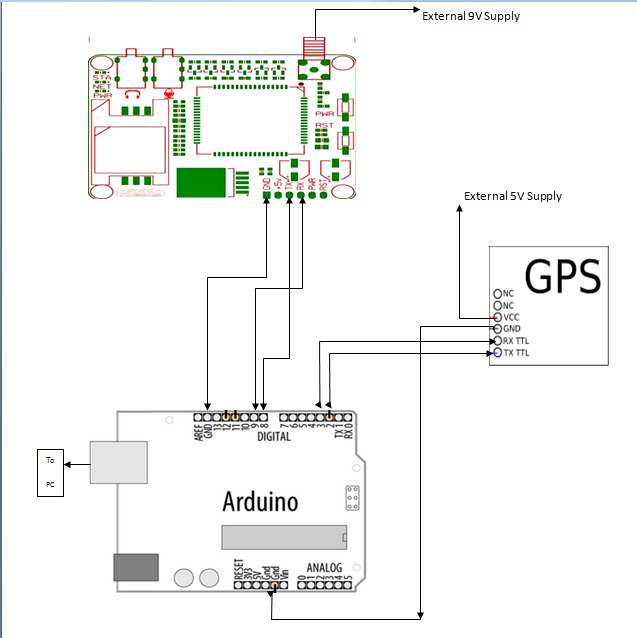
**Software Specification:**

Algorithm: Pseudo code of the program.

Platform used: Coding language used.

Flow of the program: Flowchart of the program.

**Block Diagram:**



**Hardware Specifications:**

1. **Arduino Freeduino:**

The Arduino Freeduino is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button with a stronger reset circuit. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

**Summary:**

|  |  |
| --- | --- |
| **Specifications** | **Values** |
| Microcontroller | ATmega328 |
| Operating Voltage | 5V |
| Input Voltage (recommended) | 7-12V |
| Input Voltage (limits) | 6-20V |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| Analog Input Pins | 6 |
| DC Current per I/O Pin | 40mA |
| DC Current for 3.3V Pin | 50mA |
| Flash Memory | 32 KB of which 0.5 KB used by boot loader |
| SRAM | 2 KB (ATmega328) |
| EEPROM | 1 KB (ATmega328) |
| Clock Speed | 16 MHz |

**The power pins:**

|  |  |
| --- | --- |
| Pin | Description |
| VIN | The input voltage to the Arduino board when it's using an external power source. You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin. |
| 5V | This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board |
| 3V | A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50mA. |
| GND | Ground pins |
| IOREF | This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V. |
| 0(RX) 1(TX) | Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip. |
| 3,5,6,9,10, 11 | Provide 8-bit PWM output with the analogWrite() function. |
| 10(SS), 11(MOSI), 12(MISO), 13 (SCK). | These pins support SPI communication using the SPI library. |
| Pin 13 | There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off. |
| A0 – A5 | Analog input pins. |

**Advantages:**

* Allows for serial communication on any of the Freeduino’s digital pins.
* Can be easily programmed using Arduino Software.
* The Arduino Freeduino has a resettable polyfuse that protects your computer's USB ports from shorts and over current.
* Easy to use.

**Micro Controller used in Arduino Freeduino:**

**Atmel ATmega328** 8-bit AVR® microcontrollers are high-performance RISC-based devices that combine 32KB ISP Flash memory with read-while-write capabilities, 1KB EEPROM, 2KB SRAM, 23 general-purpose I/O lines, 32 general-purpose working registers, serial programmable USART, and more. Atmel ATmega328 MCUs execute powerful instructions in a single clock cycle, allowing the device to achieve throughputs approaching 1 MIPS per MHz while balancing power consumption and processing speed. These Atmel MCUs are designed for use in industrial automation and home and building automation. The device operates between 1.8-5.5 volts.

**Features:**

* High Performance, Low Power AVR® 8-Bit Microcontroller
* Advanced RISC Architecture
* 131 Powerful Instructions
* Up to 20 MIPS Throughput at 20MHz
* On-chip 2-cycle Multiplier
* Pin Count: 28/32
* Max I/O Pins: 23
* SPI: 2
* UART: 1
* ADC: 8 channels, 10-bit resolution
* Analog Comparators: 1
* Flash (Kbytes): 32
* EEPROM (Kbytes): 1
* SRAM (Kbytes): 2
* Temp. Range: -40 to 85 °C

**Applications:**

Today the ATmega328 is commonly used in many projects and autonomous systems where a simple, low-powered, low-cost micro-controller is needed. Perhaps the most common implementation of this chip is on the ever popular Arduino development platform, namely the Arduino Freeduino.

**2. Sim900 GSM Module**

SIM900 is a Quad-band GSM/GPRS engine, works on frequencies 850MHz, 900MHz, 1800MHz and 1900MHz. It is very compact in size and easy to use as plug in GSM Module. The Module is designed with RS232 Level converter circuitry, which allows you to directly interface PC Serial port .The baud rate can be configurable from 9600-115200 through AT command. Initially Modem is in Auto baud mode. This sim900 GSM module is having internal TCP/IP stack to enable you to connect with internet via GPRS. It is suitable for SMS as well as DATA transfer application in M2M interface.

The module needed only 3 wires (Tx, Rx, GND) except Power supply to interface with microcontroller/Host PC. The built in Low Dropout Linear voltage regulator allows you to connect wide range of unregulated power supply (4.2V -13V). Using this module, you will be able to send & Read SMS, connect to internet via GPRS through simple AT commands.

**AT Commands:**

**AT commands** are used to control MODEMs. AT is the abbreviation for Attention. These commands come from **Hayes commands** that were used by the Hayes smart modems. The Hayes commands started with AT to indicate the attention from the MODEM. The dial up and wireless MODEMs (devices that involve machine to machine communication) need AT commands to interact with a computer. These include the Hayes command set as a subset, along with other extended ***AT commands***.

AT commands with a GSM/GPRS MODEM or mobile phone can be used to access following information and services:

1. Information and configuration pertaining to mobile device or MODEM and SIM card.

 2. SMS services.

3.   MMS services.

4.   Fax services.

5.   Data and Voice link over mobile network.

**Features:**

* High Quality Product
* Quad-Band GSM/GPRS 850/ 900/ 1800/ 1900 MHz
* Built in RS232 Level Converter (MAX3232)
* Configurable baud rate
* SMA connector with GSM L Type Antenna.
* Built in SIM Card holder.
* Built in Network Status LED
* Inbuilt Powerful TCP/IP protocol stack for internet data transfer over GPRS.
* Audio interface Connector
* Most Status & Controlling Pins are available at Connector
* Normal operation temperature: -20 °C to +55 °C
* Input Voltage: 5V-12V DC

**Specifications:**

* Quad-Band 850/ 900/ 1800/ 1900 MHz
* GPRS multi-slot class 10/8
* GPRS mobile station class B
* Compliant to GSM phase 2/2+
  + Class 4 (2 W @850/ 900 MHz)
  + Class 1 (1 W @ 1800/1900MHz)
* Dimensions: 24\*24\*3mm
* Weight: 3.4g
* Control via AT commands (GSM 07.07, 07.05 and SIMCOM enhanced AT Commands)
* Low power consumption: 1.0mA(sleep mode)
* Operation temperature: -40°C to +85 °C

**Specifications for SMS via GSM**

* Point to point MO and MT
* SMS cell broadcast
* Text and PDU mode

**Compatibility**

* AT cellular command interface

**3. S1315RL GPS Module.**

The S1315RL is a small form factor GPS module solution intended for a broad range of OEM products, where fast and easy system integration and minimal development risk is required. The user only need to provide DC power of 3.0V ~ 3.6V and GPS signal; the S1315RL will output navigation solution in standard NMEA-0183 protocol format.

The S1315RL features 65 channel GPS receiver with fast time to first fix and improved -148dBm cold start sensitivity. The superior cold start sensitivity allows it to acquire, track, and get position fix autonomously in difficult weak signal environment. The receiver’s -165dBm tracking sensitivity allows continuous position coverage in nearly all application environments. The high performance search engine is capable of testing 8,000,000 time-frequency hypotheses per second, offering industry-leading signal acquisition and TTFF speed.

The receiver is optimized for applications requiring high performance, low power, and low cost; suitable for a wide range of OEM configurations including mobile phone, PND, asset tracking, and vehicle navigation products.

The metal RF shielding provides protection and allows standard surface mount device pick-and-place process in fully automated assembly process; enabling high-volume, very cost-efficient production. The S1315RL is available in tape-and-reel form.

**Features:**

* 65 Channel GPS L1 C/A Code
* Perform 8 million time-frequency hypothesis testing per second
* Open sky hot start 1 sec
* Open sky cold start 29 sec
* Cold start sensitivity -148dBm
* Tracking sensitivity -165dBm
* Multipath detection and suppression
* Jamming detection and mitigation
* Accuracy 2.5m CEP
* Maximum update rate 20Hz
* Tracking current ~23mA
* Supports active and passive antenna
* Operating temperature -40 ~ +85ºC

**Technical Features:**

|  |  |
| --- | --- |
| Receiver | Type L1 C/A code, 65-channel Venus 6 engine |
| Accuracy | Position 2.5m CEP, Velocity 0.1m/sec, Time 60ns |
| Startup Time | 1 second hot start under open sky < 29 second warm start under open sky (average) 29 second cold start under open sky (average) |
| Reacquisition | 1s |
| Sensitivity - | 148dBm cold start  -165dBm tracking |
| Multi-path Mitigation | Advanced multi-path detection and suppression |
| A-GPS | Support Prompt Fix AGPS |
| Update Rate | Supports 1 / 2 / 4 / 5 / 8 / 10 / 20 Hz update rate (1Hz default) |
| Dynamics | 4G (39.2m/sec2) |
| Operational Limits | Altitude < 18,000m or velocity < 515m/s |
| Serial Interface | 3.3V LVTTL level |
| Protocol | NMEA-0183 V3.01  GPGGA, GPGLL, GPGSA, GPGSV, GPRMC, GPVTG\*1  9600 baud, 8, N, 1 |
| Datum | Default WGS-84  User definable |
| Input Voltage | 3.3V DC +/-10% |
| Input Current | 23mA tracking |
| Dimension | 15mm L x 13mm W |
| Weight: | 1.5g |
| Operating  Temperature - | 40oC ~ +85oC |
| Storage Temperature – | 55 ~ +100oC |
| Humidity | 5% ~ 95% |

**Applications:**

* PND
* MID / Net book
* Smart-Phone
* Geo-Tagging
* Automatic Vehicle Location
* Personal Tracking

Software Specification:

**ALGORITHM:**

* Set up the connections and insert the SIM card into it & the number to whom you would like to send the message to start up the device.
* As the button is pressed, the device would start working.
* Using the GPS module (in our case S1315rl GPS module), the device would track the current location and get the Latitude and the Longitude of the current location from the Satellite.
* This Latitude and Longitude would be concatenated into a single message and then passed on to the basic micro-controller (in our case FREEDUINO) as a String.
* This received message is then properly formatted and saved as a String again.
* This formatted String is then passed to the GSM device (in our case SIM 900) to be sent to a number as a message.
* The message contains a link where the receiver of the message can put in the co-ordinates included in the received message and get the exact address and the map of the area of the location.
* Thus, the location of the device can be sent to the intended person whenever you feel you are in any danger.

**PLATFORM USED:**

* **ARDUINO**
  + Arduino is a tool for making computers that can sense and control more of the physical world than your desktop computer. It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board.
  + Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs. Arduino projects can be stand-alone, or they can communicate with software running on your computer (e.g. Flash, Processing, MaxMSP.) The boards can be assembled by hand or purchased preassembled; the open-source IDE can be downloaded for free.
  + The Arduino [integrated development environment](http://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) is a [cross-platform](http://en.wikipedia.org/wiki/Cross-platform) application written in [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), and is derived from the IDE for the [Processing programming language](http://en.wikipedia.org/wiki/Processing_(programming_language)) and the [Wiring](http://en.wikipedia.org/wiki/Wiring_(development_platform)) projects. It is designed to introduce programming to artists and other newcomers unfamiliar with software development. It includes a code editor with features such as [syntax highlighting](http://en.wikipedia.org/wiki/Syntax_highlighting), [brace matching](http://en.wikipedia.org/wiki/Brace_matching), and automatic indentation, and is also capable of compiling and uploading programs to the board with a single click. A program or code written for Arduino is called a "sketch".
  + Arduino programs are written in [C](http://en.wikipedia.org/wiki/C_(programming_language)) or [C++](http://en.wikipedia.org/wiki/C%2B%2B). The Arduino IDE comes with a [software library](http://en.wikipedia.org/wiki/Software_library) called "[Wiring](http://en.wikipedia.org/wiki/Wiring_(development_platform))" from the original Wiring project, which makes many common input/output operations much easier. Users only need define two functions to make a runnable [cyclic executive](http://en.wikipedia.org/wiki/Cyclic_executive) program:
* setup(): a function run once at the start of a program that can initialize settings
* loop(): a function called repeatedly until the board powers off

**FLOWCHART:**

**APPLICATIONS:**

* Collar belt of the Pet

The device can be embedded in the collar belt of the pets and the pets can be set out free. Through the device we can track our pet as to where he is currently so as to get it back home.

* Vehicle Tracking

The device can be embedded into a vehicle. If the vehicle is borrowed by anyone from you or you have parked it somewhere and want to check out where it is then you can you can use get the location of it using the device on your phone.

* Security of important belongings if stolen

The device can be fit in to any of the important personal belongings. If the belonging gets stolen, then you can get to know its location through the device and can get it back.

* Get your own location if lost at a place like Jungle

If you are an adventurer and you get lost into the jungle by any chance, then using this device you can get your current location and the map or way to get out of that place and find the exact route.

* To keep your personal things secure like Lockers

The device can also be used to keep lockers safe by embedding it to the locks of the lockers when you are not in the house or near your locker and want to keep it away from others. As soon as the lock gets open the button gets pressed and you may get to know that the locker is opened up and you can take an action on it.

**Future Scope of the Project:**

* The accuracy of the location can be increased by implementing some advanced logics on calculating the location.
* Camera can be attached to the device for certain more application utilities can safety of lockers.
* A server web page can be designed for police administration so as to help them get there faster.
* An LED display can be attached with the device so as to show the location to the user himself.

**References:**

URL’s:

* http://arduino.cc/
* http://www.hughes.com/AT\_Command\_Reference.html#AT\_plusCSMP
* http://www.open-electronics.org/using-the-gsmgprs-gps-shield-call-examples/
* http://www.elecfreaks.com/store/4-frequency-ultra-gsmgprs-module-sim900-p-109.html
* http://forums.adafruit.com/viewtopic.php?f=25&p=192988
* http://www.instructables.com/id/Connecting-GPS-module-to-Arduino/
* http://blog.oscarliang.net/arduino-gps-tutorial/
* http://www.doctormonk.com/2012/05/sparkfun-venus-gps-and-arduino.html
* http://arduiniana.org/libraries/tinygps/
* https://www.inkling.com/read/arduino-cookbook-michael-margolis-2nd/chapter-6/recipe-6-14
* http://www.geeetech.com/wiki/index.php/Arduino\_GPRS\_Shield
* http://arduino.cc/
* http://www.macrogroup.ru/content/data/store/images/f\_1072\_5364\_1.pdf
* http://tronixstuff.com/tutorials/
* http://www.jeremyblum.com/category/arduino-tutorials/

Books:

* A complete beginner’s guide to arduino by Mike McRoberts.