Design and Implementation of an Automobile Theft Deterrent System

Ajay D. Murthy[1], B. N. Shobha[2], Krishnamurthy Vaidyanathan[3]
[1]M.Sc. (Engg.) student, [2]Assistant Professor, [3]CEO EI Labs
Embedded Systems Design Centre
M.S. Ramaiah School of Advanced Studies, Bangalore 560 054

Abstract

Medium and heavy duty trucks transport valuable or hazardous cargo which makes these vehicles targets for thieves and anti social elements. A massive effort has been made in research and development of security systems for automobiles. In European nations the insurance companies have made anti theft systems mandatory for automobiles. Many automobile theft deterrent systems have been designed to interrupt the ignition or a hidden switch to be depressed. These techniques are easy to overcome. Deactivation technique is a new and unique concept to be introduced in the automotive electronic sector for an Automobile Theft Deterrent System (ATDS). The deactivation technique is based on interrupting the ground connection to the Power Distribution Centre (PDC) by providing an activation message from the users' mobile unit. Once the PDC is disconnected the Auto Shut Down Relay (ASD) is latched, this in return switches OFF the engine and the complete electrical units present in the vehicle. Further the vehicle is tracked through GPS. The GPS data is written to the company database and transferred to the GIS to provide the user with a graphical locator.

Key Words: ATDS, GSM, GPS, Deactivation technique, Auto Shut Down Relay

Abbreviations

ARM	Advanced RISC Machine
ASCII	American Standard Code for Information Interchange
BDM	Background Debug Mode
CDMA	Code Division Multiple Access
CPU	Central Processing Unit
DGPS	Differential Global Positioning System
ELF	Executable and Linkable Format
EPI	Embedded Performance, Inc.
GIS	Geographic Information System
GLONASS	Russian GLObal NAvigation Satellite System
GND	Ground
GNSS	Global Navigation Satellite System
GPIO	General Purpose Input Output
GPS	Global Positioning System
GSM	Global System for Mobile communication
IDE	Integrated Development Environment
IVU	In Vehicle Unit
KB	Kilo Byte
LCD	Liquid Crystal Display
Kmph	Kilometres Per Hour
MB	Mega Byte
NAVSTAR	NAVigation System with Timing And Ranging
NMEA	National Marine Electronics Association
PVT	Position Velocity Time
RAM	Random Access Memory
RISC	Reduced Instruction Set Computer
RS232	Recommended Standard 232
SD	Secure Digital
SDRAM	Synchronous Dynamic RAM
SOC	System On Chip
STUART	Standard UART
TXD	Transmit Data
UART	Universal Asynchronous Receiver Transmitter
ATDS	Automobile Theft Deterrent System

1. INTRODUCTION

The Automotive telematics market has shown a growth in recent years. The security systems for automobiles have also shown a vast improvement. Actually, today's newer cars with their on-board computers or electronic control modules (ECMs) make it easier to diagnose performance problems than ever before. The same ECMs are complex and vary in design, several vehicles like the Ford and GM carry more than one ECMs.

The ATDS provides two services, that is

a. **Vehicle tracking:** The tracking of vehicles that leads to the collection and transmission of vehicle position and its other information to base station. This data is delivered to the company's database which in return is taken to the GIS system. The GIS contains the vehicle location in precise and might carry additional features like the fuel available, drivers identification, vehicle speed etc.
b. **Vehicle Immobilization:** The vehicle movement is disabled by raising a message through the GSM network. By deactivating the vehicle the complete electrical and electronic systems in the vehicle are made not to function. The vehicle immobilization is done based on parameters like the vehicle speed and the ignition key status.

1.1 GPS Overview

The GPS is a gift from the US DoD; it consists of a structure of 24 NAVSTAR satellites orbiting the earth providing realtime data to find the location and several

other information like the course over ground (COG), Speed over ground (SOG) etc. The system works through a DSP based GPS receiver attached to the embedded system, the lat, long, etc are calculated via the inbuilt DSP algorithm available with the GPS receiver.
The GPS message is a continuous stream of data transmitted at 50 bits per second. Each satellite relays the following information on earth:

- System time and Clock correction values
- Its own highly accurate orbital data
- Approximate orbital data for all other satellite

In the ATDS design, ublox from Atmel is incorporated to perform the GPS related events and to provide the Wavecom Q24 module with relevant data like the lat, lon, speed etc. The GPS ublox communicates with the Q24 plus through anUART, the ublox has its own input and output message format and the flow control mechanism is highly efficient. The general data out of a GPS receiver is called as the NMEA strings, the flow of the NMEA strings is continues when the module is plugged in to the host.
The wavecom module has two UART's, one UART is connected to the GPS receiver and the other UART is available for debugging. In our design the NMEA is not parsed by the application, the application raises a data request through the ublox proprietary input messages. These messages are provided to the ublox which in return provide the required data only when a request is made. The messages start with a "$PUBX" string and the data is completely in ASCII format.

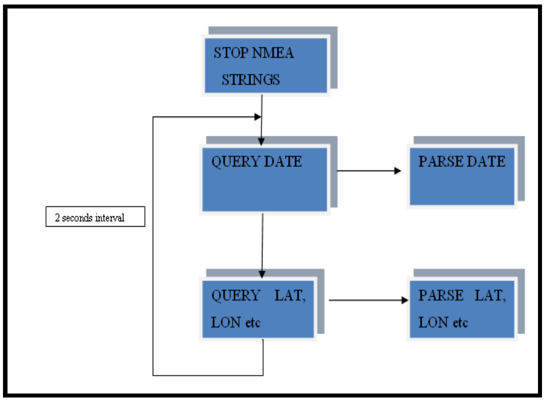

Fig. 1.1 GPS System Design

1.2 GPS Design Model

1 The NMEA strings are stopped
2 The Application raises a query at every 2 seconds interval
3 Each data string begins with a '$' and ends with a carriage return/line feed sequence and can be no longer than 80 characters of visible text (plus the line terminators).
4 The data is contained within this single line with data items separated by commas.
5 The data itself is just ASCII text.
6 Program that read the data use the commas to determine the field boundaries and not depend on column positions.
7 The interface speed is 9600 baud with 8 bits of data, no parity and one stop bit.
8 Among the NMEA data sent by the GPS receiver only GPGGA data is parsed.
9 The GPS data is read from UART2 of the ATDS board.
10 Flow control mechanism is provided.
11 The output of parsing the $PUBX data is latitude, longitude, HDOP, VDOP, N S Indicator, E W Indicator etc.
12 The latitude and longitude data has to be output to a text file for navigation application and GSM application

The Fig. 1.1 demonstrates the working of the GPS system for the ATDS.
The GPS ublox unit provides continues NMEA string, thus initially when the ATDS is switched ON, the NMEA strings are to be stopped. The set of commands supplied to the ublox are as follows.
$PUBX,40,GGA,0,0,0,0*5A
$PUBX,40,GSA,0,0,0,0*4E
$PUBX,40,VTG,0,0,0,0*5E
$PUBX,40,RMC, 0,0,0,0*47

2. DESIGN OF SMS APPLICATION

Open AT has a set of API's for the SMS functionality. Initially for the module to handle any sort of SMS related function a subscription has to be carried out. Once the subscription is done a set of handlers are provided i.e. the data handler and the control handler. The data handler is invoked when an incoming message is encountered.
The functions included in the SMS application file are:

- **s8 adlSmssubscribe(adl_smshdlr_f SmsHandler,adl_SmsCtrlHdlr_f SmsCtrlHandler,u8 mode);**

This prototype subscribes to the SMS service in order to receive SMSs from the network.

- **Bool SMS_Data_Handler (ascii *SmsTel, ascii *Time, ascii *SmsText)**

This data handler is invoked each time an SMS is received from the network, this contains the originating phone number, the time and the text

The control handler is as defined below:
typedef void (* adl_smsCtrlhdlr) (u8 Event, u16 Nb);
The control handler indicates the current event i.e. the state of the message.
The indication is carried out by providing any of the three response statement [11] i.e.

- ADL_SMS_EVENT_SENDING_OK

This is to indicate that the SMS was sent successfully

- ADL_SMS_EVENT_SENDING_ERROR

This indicates that an error occurred during the message sending process.

- ADL_SMS_EVENT_SENDING_MR

This indicates that the message was sent successfully and the Nb carries the message reference value.

The application carries the SMS functions as described:

1. EILMN

This message is used to set the master's phone number, this is done to maintain authentication. This avoids others apart from the master to gain access to the system. Ex EILMN+9198451XXXXX, once this is sent to the ATDS system, the number 98451XXXXX is stored as the master's phone number and thus further the activation and the deactivation codes are accepted only from this number.

2. EILACSET

This message is used to set the activation code and this code can be changed any number of times, this message to set the activation code will be accepted only from the master's number. Ex EILACSET1234, once this message is received the system stores the number 1234 as the activation code and further the activation can be carried out using this code.
Note: This message won't be accepted when the ATDS is in the Active state.

3. EILDCSET

This message is used to set the deactivation code and this can be set any number of times, this message employed to set the deactivation code will be accepted only from the master's number. Ex EILDCSET4321, once this message is marked, the system stores the final four digits 4321 as the deactivation code and further can be used to deactivate the system.
Note: This message will not be considered by the system when it's under Active state.

4. EILACT

This message is used to switch the system to Active state and this message is accepted only from the master's number and once the message is received the received code is verified with the stored code.

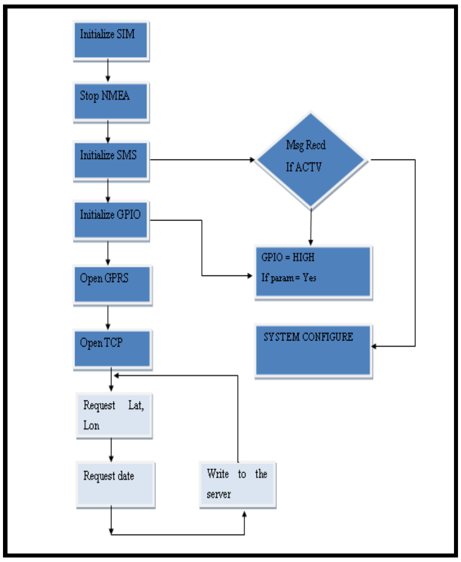

Fig. 3.1 ATDS Design Overview

Once the result proves TRUE, the system is set to Active state. Ex EILACT1234, this puts the system into Active state.

1. EILDCT

This message is used to switch the system to Deactive state and this message is accepted only from the master's number and once the message is received the received code is verified with the stored code. Once the result proves TRUE, the system is set to Deactive state. Ex EILDCT4321, this puts the system into Active state.

If an MP is unable to detect any neighbor MPs, it shall adopt a Mesh ID from one of its profiles, and proceed to the active state, which, in the case of a MAP is the access point initialisation state. This will occur when the MP is the first device to power on (or multiple MPs power on simultaneously). Any peer MP links will be established later as part of the continuous mesh formation procedures. Each entry shall contain the information shown in Table 7.

3. ATDS DESIGN OVERVIEW

The ATDS is incorporated with a MEMS accelerometer to check the vehicle speed. Once the ATDS is set to active mode, the ATDS monitors the Ignition key status and the vehicle speed. The PDC and the ground connection is interrupted only if the ignition key is not in the RUN position and the vehicle speed to be zero. Once these two conditions are satisfied, the relay connected between the ATDS and the PDC is toggled. The working is as described in the flow chat in figure 3.1.

In the above flow chart the param describes the set of parameters under which the system will be put the Active Mode. The wavecom module has 5 GPIO and can be configured by the user, once the parameters are satisfied the relevant GPIO gets a HIGH thus the PDC and the ground connection is discontinued. This brings the vehicle to a complete halt, immobilized. The obtained data like the speed value from the MEMS, lat, lon, COG, N-S Indicator, E-W Indicator, HDOP,VDOP etc are set is format which is acceptable by the DNS server. The format is as given below:
"GET trackit/urlupdate.asp?DATA HTTP/1.1\nHost: xxx.xxx.xxx.xxx"

4. PORTING THE EXECUTABLE FILE ON THE WAVECOM MODULE

Once the code design is completed using the Visual C++ or Eclipse, the code is compiled and built. This Open AT IDE is designed to generate a gcc folder in the destination folder. The gcc folder contains a set of files with variable extensions, the file with wpb.dwl extension is the executable to be taken to wavecom core module.
The code is transferred to the wavecom through the Hyper Terminal.
The steps followed are as given:
"AT+WOPEN=2"
This command lists the OS present and the application software ID.

"AT+WOPEN=3"
This command clears the flash memory of the wavecom module
"AT+WOPEN=4"
This command erases the code present on the wavecom memory, now the system is ready to accept a new executable file.
To transfer the file at a faster rate to the wavecom module, the baud rate of the wavecom module is set to 115200 by providing "AT+IPR=115200". Once this is done the baud rate of the hyper terminal is set to 115200 through the properties section.
To download the file to the wavcom module "AT+WDWL" is issued and the file is transferred using the send file option in the hyper terminal. After the completion of this, the system is reset by providing "AT+CFUN=1". This command resets the module at the system will set back to 9600 baud rate, thus the hyper terminal is also set to 9600 baud rate and further to RUN the application, "AT+WOPEN=1" is issued.

5. VALIDATION AND DISCUSSION OF RESULT

The ATDS was tested in stages, initially the API's for the UART and the GPS functions were designed, these unit tests were carried out and initially for precession the received latitude and the longitude were tested on maps.google.com. The result is as shown in fig 5.1.
The later stage of testing was after the completion of the GPRS API's design, the output screen is as shown in fig 5.2. The figure shows the system behaviour during initialization of the ATDS, the NMEA stop strings being provided to the ublox etc.
The first set of data is the NMEA stop strings, the second output data is to indicate the SIM OK status, the third data is a set of GPRS activation and status strings. Once these conditions are achieved the system raises a query to the GPS receiver to provide required data, the query strings are followed by the $PUBX string.
The received data is parsed and written to the global structure.
The data from the global structure is written to the server. The successful data write operation is notified through the return integer value from the server. Once the integer value and the number of bytes tally, the system is considered to have written the data successfully.
Further when the system is set in Active mode, the ATDS monitors the two parameters i.e. the ignition key and the vehicle speed continuously. Once these two parameters are satisfied it sends a system activation message on the serial port, this message can be captured on the hyper terminal which is as shown in figure 5.3. The figure 5.3 as well describes the system deactivation mode.
The marking 1 in figure 5.3 depicts the ATDS in ACTIVE mode followed by the masters phone number and relay deactivated message. Further the second message mentions the deactive stage and followed by the masters phone number.
The vehicle location is viewed on the company's GIS map, this is as shown in the figure 5.4. The vehicle location is indicated by a car image as shown in figure 5.4. Once the vehicle is set in active mode, the vehicle is tracked through this platform.

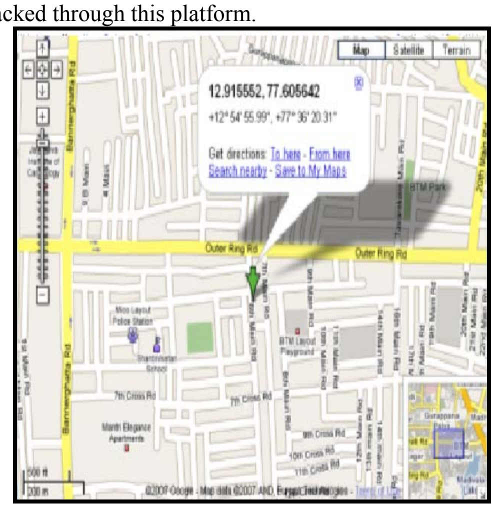

Fig. 5.1 GPS Data Verification on Google Maps

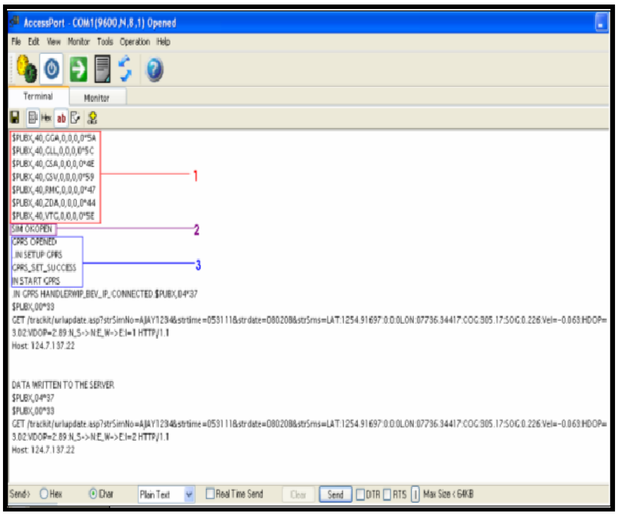

Fig. 5.2 ATDS Start-up

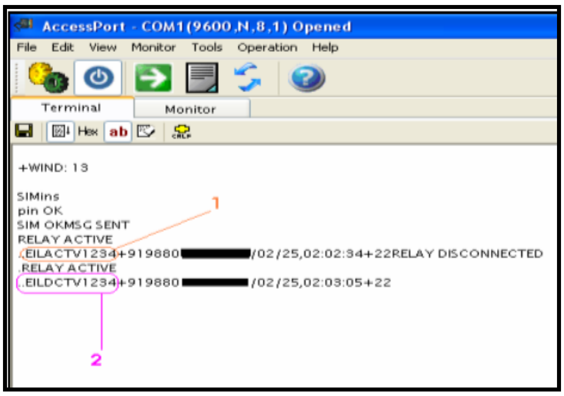

Fig. 5.3 ACTIVE and DEACTIVE Mode

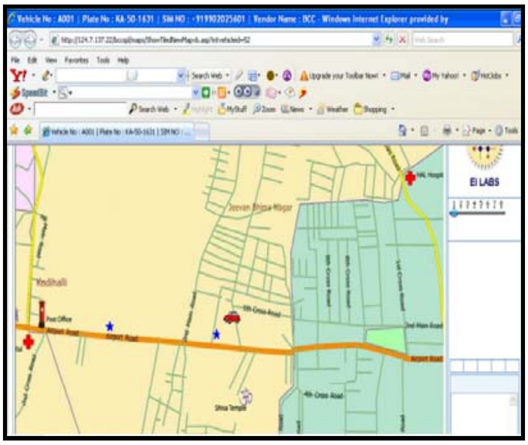

Fig. 5.4 Vehicle Tracking Using EILABS GIS Platform

6. CONCLUSIONS

The project has given a great insight into the concepts of creating Real Time Embedded systems. The Automobile Theft Deterrent System achieved a satisfactory level of operation.

Based on the design and analyses presented in the previous sections, the following conclusions can be drawn:

1. GPS antennas which can pick up precise signal even when the system is indoor will add great value to the system, in terms of tracking and accuracy.
2. Fuel level checking sensor can be add on which can be used to avoid fuel theft as well.

The vehicle location is highly precise, the user gets the exact location of the vehicle. The speed check is carried out through the MEMS sensor as well verified through the GPS data from the ublox.

REFERENCES

[1] http://products.thalesnavigation.com/en/products/aboutgps/howworks.asp

[2] http://www.thalesnavigation.com/en/products/aboutgps/glonass.asp

[3] http://www.thalesnavigation.com/en/products/aboutgps/galileo.asp

[4] http://www.thalesnavigation.com/en/products/aboutgps/augmentation.asp

[5] Global Positioning System. (n.d.). AsianGPS. Retrieved April 10, 2002, from http://www.asiangps.net/tutorial/

[6] Mishra, P. (2001). The National GPS technology initiative. Retrieved Mar 11, 2002, from http://www.gisdevelopment.net/technology/gps/techgp0010.htm

[7] Mullings, W.K. Lindsey,"*An application of autonomous computing for the automotive industry*", IEEE Consumer Electronics, P 98 – 99, 17-19 June 2003

[8] Yuanchun Li; Guangjun Liu; Xiao Zhou,"*Fuel-injection control system design and experiments of a diesel engine*",, IEEE Control Systems Technology, PP 565 – 570, July 2003

[9] Micheal Bhaezel, pen A.T., "*The Advantages of the OpenAT software suite"* Wavecom OpenAT white paper, October 2006

[10] The Automotive Hand book, Robert Bosch, Mar 2000

[11] Wavecom ADL user guide, Wavecom corporation, November 2006

[12] AN2003 "Application note:LD302DL 3-axis accelerometer", www.st.com/app/an2003, (Online September 2007)