Design and Implementation of a GPS based Personal Tracking System

Sudhanshu Janwadkar[1], Dipak Bhavar[2] and M.T. Kolte[3]
[1]MIT College of Engineering, Pune, India
[2]Capgemini India Pvt. Ltd., Pune, India
[3]Department of E&TC, MIT College of Engineering, Pune, India
E-mail: [1]shudhanshu.janwadkar@gmail.com

***Abstract*—Tracking based applications have been quite popular in recent times. Most of them have been limited to commercial applications such as vehicular tracking (e.g tracking of a train etc). However, not much work has been done towards design of a personal tracking system. Our Research work is an attempt to design such personal tracking system. In this paper, we have shared glimpses of our research work.**

The objective of our research project is to design & develop a system which is capable of tracking and monitoring a person, object or any other asset of importance (called as target). The system uses GPS to determine the exact position of the target. The target is aided with a compact handheld device which consists of a GPS receiver and GSM modem. GPS receiver obtains location coordinates (viz. Latitude & Longitude) from GPS satellites. The location information in NMEA format is decoded, formatted and sent to control station, through a GSM modem. Due to use of OpenCPU development platform, no external Microcontroller is required, with additional advantage of compact size product, reduced design & development time and reduced cost.

Thus, the proposed system is able to track the accurate location of target. This system finds applications in tracking old-age people, tracking animals in forest, tracking delivery of goods etc. Our final designed system is a small-size compact 1.5''X3.75'' Tracker system with position accuracy error <30m (100 feet).

Keywords—GPS; GSM; Tracking; Navigation; NMEA; AT Commands; OpenCPU

I. Introduction

In recent times, tracking has become an important issue. Failure to keep track of assets is being related to loss of money and time. Tracking, in our sense, is to determine the pin-point location of the parcels & goods, animal or person of interest, at all times. With this broad definition in mind, we shall establish both ethical and economic importance of tracking in next few lines.

In the current scenario, tracking of goods and parcels is done exclusively through bar-code system. When the item is packed, a barcode sticker is pasted along with the packaging of the item. As the parcel is carried away along its trajectory to the destination, at every check point in the system, the bar-code is scanned manually and the position of the parcel is entered in the system. However, this approach has two severe concerns. First one being, due to human intervention in the system, the system immensely depends on the enthusiasm of the human operator to make an entry in the online database. Secondly, all the stake-holders know only about the position of the parcel at the checkpoint, but nowhere in between.

Similarly, in case of endangered wild life animals, the tracking currently depends on their vision captured in high-resolution cameras. However, the exact and accurate position is known pnly at unpredictable instants. In case of causalities of the animal, the same may not be known for a long time.

Tracking of human beings has been a matter of interest too. Especially for the old age people who need special attention, tracking becomes important. Our Personal Tracker is customized for this particular application, although it can be used in other Tracking applications as well.

GPS based trackers have been integrated into public transport systems and they are popular in use. However, tracking of individual person or assets, as discussed has not been made possible yet. We firmly believe, this is because none of the systems has been robust and diverse enough to serve as a solution to each of the issue discussed above.

Our approach based on OpenCPU platform is a radical solution for personal tracking applications. Based on OpenCPU platform, we have implemented a compact Personal tracker 1.5'' X 3.75'' inches in dimensions, which is capable of determining the exact position of the target with an accuracy< 30m(CEP).

II. Objectives

Understanding that our system will be used in critical applications where accuracy and size, both, matter a lot, we have objectives as:

1. To design a highly accurate position tracking system
2. To keep the size of device as small & compact as possible and keep weight low.

Low size and weight objectives ensure that the system doesn't interfere with the daily activities of the person using it.

III. LITERATURE SURVEY

As discussed in previous sections, the concept of tracking systems is not new.

As we go behind in chapters of history of space based navigation systems, GPS project was the first launched by US Department of defence in 1973, initially with 24 satellite aim. Currently, there are 31 satellites used by GPS. With Russian Glonass, Chinese BeiDou, Indian IRNSS and EU's Galileo, there are multiple such satellite based systems. More details about development of GPS system can be referred in [1],[2] and [3].

In [4], E. Hammerle , P. Winton and S. Fett, propose an indoor position tracking system. This system tracks the position using estimation technique known as Partial Pulse Positioning, which makes use of a radio-frequency (RF) transmitter. Based on the reflections obtained from different reflectors placed in the indoor-environment, the position of the RF transmitter is determined. In [5], F. Evennou and F. Marx present an aided dead-reckoning navigation system based on WiFi signal strength measurements. The position and attitude are determined using low cost sensors. The errors are then fixed using WiFi signal strength measurements. Signal processing algorithms are used to correct the drift. Inertial navigation systems interact with the WiFi positioning system to provide highly-accurate real-time navigation .

A common issue with GNSS/GPS based navigation system is multipath propagation. The issue is especially serious in urban areas with tall buildings. It is fairly possible that the direct line of sight to a satellite may be blocked by a building or skyscraper and its signal might reach the receiver on the ground, only after one or more reflections. Since the signal path is longer for the reflected signal, ranging errors occur, which result in false prediction of position. The issue of reflection by buildings is also possible even when there is direct line of sight. Meguro *et al.* [6] propose a precision positioning technique which can be applied to track vehicles in urban areas. The proposed technique mitigates multipath propagation with help of an omni-directional infrared (IR) camera. Positioning is performed only considering the visible satellites that have lesser multipath errors. Invisible satellites (satellites which are hidden behind buildings)are not considered, although signals are received from all satellites. The IR camera aids in judging which signals are to be considered for positioning.

Kouji Murakami *et al.* [7], have proposed a system for tracking of everyday objects for a service robot. The system consists of an intelligent cabinet, floor sensing system and a data management system. Using a RFID reader, the intelligent cabinet can measure the position of an object on itself. The floor sensing system uses a laser range finder. Using laser range, it is capable of measuring the position of an object on the floor and the position of a human walking in a room. The captured data is logged in by data management system for providing inputs to service robot.

In case of multipath propagation, removing the reflected signals might lead to a shortage of satellites in fixing the position. S Tang *et al.* [8], propose cooperative relative positioning (CoRelPos) scheme. In this scheme, Correlated information, including that of reflected signals, is used to compute relative position.

M Fuzi *et al.* [9], propose a method to estimate the location of pedestrians by locating their moving mobile devices. GPS is used in order to determine the absolute position of the mobile device & dead-reckoning methods using built-in sensors in the mobile are used to obtain the relative movement of the mobile.

IV. OPENCPU DEVELOPMENT PLATFORM

In a traditional GPS based tracking system, the GPS coordinates by a GPS module would be read by an external Microcontroller and the GPS data would be processed and sent to database or user using GSM/GPRS/Bluetooth etc. However, the complexity of such a system is simplified using OpenCPU platform. OpenCPU Core System is a combination of hardware and software of GSM/GPRS module. It has built-in ARM7EJ-S processor and is built over Nucleus Operating System.

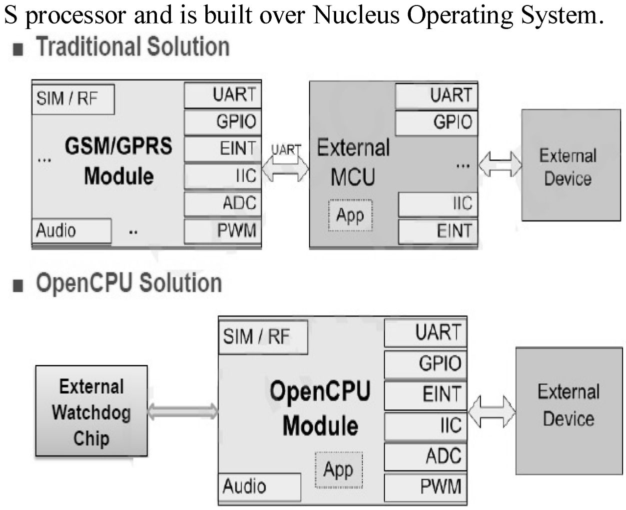

Fig. 1: Illustration of OpenCPU Solution

OpenCPU is an embedded development solution for M2M field applications where GSM/GPRS module can be designed as the main processor. It helps to facilitate the design and accelerate the application development. OpenCPU makes it possible to create applications and embed them directly into GSM/GPRS modules to run without external Microcontroller.

A. *Advantage of using OpenCPU Platform*

Compared to the traditional approach, OpenCPU solution makes hardware design easier. The difference between the traditional approach and OpenCPU based approach to same solution is shown in Fig. 1.

Thus, the advantages of using OpenCPU based approach include:

1. Simplify circuit design
2. Decrease the product's size
3. Reduce product development time
4. Decrease the total cost and power consumption

B. *OpenCPU Architecture*

The architecture of OpenCPU consists of 3 layers: Core system, RIL layer and Application Layer. This has been shown in Fig. 2.

Core System: The Core System of OpenCPU is combination of hardware and system software of GSM/GPRS module. Hardware consists of built-in ARM7EJ-S processor and peripherals such as Watchdog Timer, ADC, PWM etc. It has been built over Nucleus operating system.

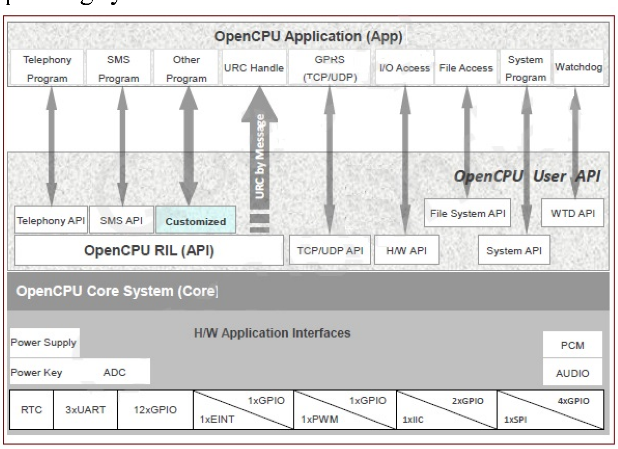

Fig. 2: OpenCPU Architecture Layers (Image Courtesy: Quectel)

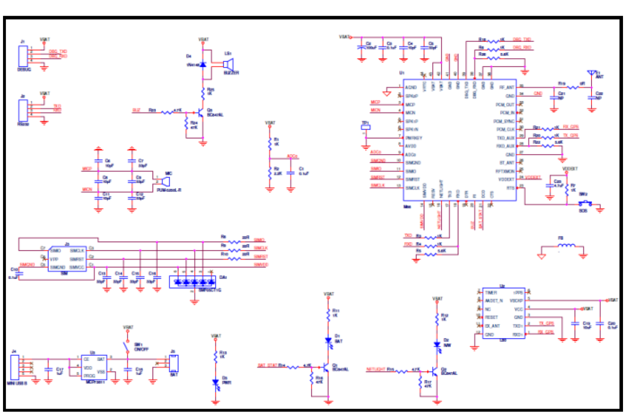

Fig. 3: Layout of Proposed Tracker System

OpenCPU RIL Layer: OpenCPU RIL Layer, is a open source layer which is embedded into user API layer. Using RIL layer, developer can simply call API to send AT commands and immediately get the response when API returns. Some of the basic API such as telephony, SMS etc are provided along with OpenCPU package. However, the developer can write his/her own custom API or new APIs for different functionalities.

Application Layer: In Application Layer, the desired functionality is realized by calling suitable API from RIL layer. The firmware is developed for Application layer.

Detailed description of OpenCPU is available in OpenCPU Documentation [10].

V. EXPERIMENTATION DETAILS

The entire firmware is written in Embedded C language in Eclipse CDT. Arm Sourcery CodeBench Lite tool chain is required for OpenCPU platform. Details about Eclipse CDT & Sourcery CodeBench can be referred at Eclipse Website [11] and MentorGraphics website respectively [12].

In our Research work, we have used Quectel L80 GPS module and Quectel M66 GSM/GPRS module. The GPS L80 module is interfaced to GSM/GPRS Module through UART pins. The Layout/Block diagram of our system is as shown in Fig. 3.

A. *Features of our Personal Tracker*

As described earlier, although tracker can be used for any of the personal applications, the features have been customised for its use in tracking of elderly persons.

1. Battery operated system: To keep the Tracker portable, The entire system has been powered using 3.3V output, 1100mAh battery. There are features to indicate low battery condition, to notify the user to charge the battery. The battery can be charged by using USB charging cable.
2. Emergency SOS switch: In case the user is in emergency (say an elderly person using the tracker has sudden health ailment), he/she can press the SOS switch and an alert call will be initiated. There are up to 2 telephone/mobile numbers to which telephonic call will be dialed, so that user can convey his message.
3. Update Emergency number through SMS: The telephone number to which alert call has to be sent in case of emergency can be updated using SMS. In our Tracker, sending SMS 'UPDATE *123# <number>" to the SIM inserted in the tracker will cause update of Emergency number. We have used string parsing for the module to understand the correct SMS.
4. Tracker Locator: In order to locate the Tracker (in case it is misplaced), a buzzer has been provided. Dialing the SIM card inserted in the Tracker, the buzzer will ring and Tracker can be located.

B. Working

When the Tracker is turned ON using the power switch, it waits till the SIM is registered to the GSM & GPRS network. LED blink pattern 1(Blink 5 times, 500ms each) indicates successful registration.

At same time GPS L80 module is also turned on and it starts sending GPS information in NMEA format[13]. The GPS information consists of 7 strings. Of our importance is, $GPGGA string as it contains timestamp and location information in terms of Latitude and Longitude information. $GPGGA string has been explained in next subsection.

GPS information is read through UART protocol and necessary string parsing is done to get the information in an format easy to interpret by human operator.

At fixed intervals of time(At every 30 minutes in our code), this GPS information is sent to a Email Id Registered by the user using SMTP protocol. Details of SMTP protocol are mentioned in [14],[15].

The battery voltage is monitored by ADC peripheral of the module. If it goes below decided threshold (1.9V in our code), LED blink pattern 2(Blinks 7times, 250ms each) indicates Low battery condition.

At any instant of time, if user presses SOS button (switch), a telephone call is initiated to the number set by the user. If the first call is failed, immediately another telephone call is initiated to second number registered by user. Microphone is included so that target can pass his message. The process of updating number to which emergency call should be dialed is explained in previous subsection

At any moment, if a call is made to the SIM inserted in the module, the buzzer rings and is used to locate the tracker.

C. $GPGGA NMEA Format

TABLE 1: $GPGGA NMEA FORMAT

Name	Example Data	Description
Sentence Identifier	**$GPGGA**	**Global Positioning System Fix Data**
Time	170834	
Latitude	18.5204, N	
Longitude	73.8567, E	
Fix Quality: - 0 = Invalid - 1 = GPS fix - 2 = DGPS fix	1	Data is from a GPS fix
Number of Satellites	5	5 Satellites are in view
Horizontal Dilution of Precision (HDOP)	1.5	Relative accuracy of horizontal position
Altitude	560.2, M	280.2 meters above mean sea level
Height of geoid above WGS84 ellipsoid	-34.0, M	-34.0 meters
Time since last DGPS update	blank	No last update
DGPS reference station id	blank	No station id
Checksum	*75	Used by program to check for transmission errors

D. Steps for Programming the OpenCPU

1. After the module is powered on, RIL Layer will send MSG_ID_RIL_READY message. Keep checking for this message and once received, call Ql_RIL_Initialize function to initialize RIL layer. This pre-defined function sends AT commands for initialization.
2. All the status change of GSM module are conveyed using Unsolicited Result Code(URC) messages. All tasks such as SIM being registered to GSM network, incoming call indication or result of sending AT Command such as Success or Failure along with cause are communicated using URC code only. Next step is to keep checking for URC indications. This is conveyed by the RIL message MSG_ID_URC_INDICATION.
3. Write the API for required functionality. This might include customising existing API for our application or writing a new API. All APIs reside in RIL layer & their prototypes are defined in header files.
4. Write the CallbackHandler. This is an optional step. The API will send the AT Command. If some processing needs to be done based on result received for the AT command, then that code has to be written in CallbackHandler.
5. Keep checking the RIL messages and call API & their CallbackHandler appropriately.

The battery output voltage level is monitored using ADC0 of the M66. The SOS switch, Microphone, Buzzer, Network LED (blinks according to strength of GSM network, blinking fast indicates strong GSM network) and Status LED (used to give Low Battery indication, incoming call indication etc) are all connected to GPIO pins of M66. The GPS L80 and M66are connected through Rx, Tx pins in criss-cross manner. SIM card interface is provided to slot for the micro-SIM card. The pin connections are shown in Figure 3.

E. Algorithm

```
//step 1: Initialise all GPIO pins (Includes setting direction, level)
Ql_GPIO_Init(PINNAME_RI,PINDIRECTION_OUT, PINLEVEL_ LOW, PINPULLSEL_PULLUP);
..
//Keep all unused pins to low to conserve power
// Step 2: Check battery voltage level, call API for ADC
RIL_ADC();
// Initialize ADC (sampling count, sampling interval)
Ql_ADC_Init(adcPin, 5, 200);
// Start ADC sampling
Ql_ADC_Sampling(adcPin, TRUE);
..
//API will send AT+QADC to get value, processing is done in //CallbackHandler
```

```
Callback_OnADCSampling(Enum_ADCPin adcPin, u32 adcValue, void *customParam)
{
if( *(adcValue) >= 1200)
{
GPIO_LEDBlink1();
}
}
 *((s32*)customParam) += 1;
}
..
//Step 3: Start Timer
//QI_Timer_Start(u32 timerId, u32 timerValue, bool auto-repeat
QI_Timer_Start(Timer1, u32 1800000, TRUE);
//Process timer Interrupt in its CallbackHandler)
Callback_Timer(u32 timerId, void* param)
{
if(timerValue>18000000)
UART_GPS();
GPS_stringParsing(*string);
sendGPS_SMTP(mail, *string);
}
//Step 4: check if SOS switch is pressed
level=QI_GPIO_GetLevel(pinName RI)
if(level==0)
Dial_phn1();
//Step 5: Check messages from RIL layer & call API if required
case MSG_ID_URC_INDICATION:
sys.GetOSMsg(msg)
case MSG_ID_RIL_READY: ret =
Ql_RIL_Initialize();
 break;
case MSG_ID_URC_INDICATION:
switch (msg.param1)
{
case URC_SIM_CARD_STATE_IND:
....
case URC_GSM_NW_STATE_IND:
....
case URC_COMING_CALL_IND:
GPIO_buzzer();
```

VI. Results and Conclusion

We have tested our module in indoor environment which is top floor of a building, where most GPS satellites should be in direct line of sight. The GPS location obtained was compared with that obtained using PowerGPS software and also Google Maps. We were successful in determining the precise location with error of unit second of longitude/latitude at worst. This roughly approximates to 100 feet or 30m, i.e. maximum position error was less than 30m. Also, we observed that depending on strength of GPS/GPRS network, there was a delay of 6-30s in initiating the telephone call, sending mail etc. This delay can be minimized by choosing Network operator with good coverage. Our designed tracker is compact in size and with position accuracy error <30m, our objectives are successfully attained.

VII. Scope For Future Work

A Map-based application for Android or iOS can be developed and provided along the Tracker. Instead of getting location longitude and latitude coordinates, which are difficult to understand by common user, the location can easily be viewed on a Map.

Work needs to be done to improve accuracy of position fix and improving performance in indoor environment.

Acknowledgement

We are thankful to Capgemini India Pvt Ltd for the facilities and infrastructure provided. We express special gratitude to Ms Rutuja Kulkarni, Ms Smita Khatavkar, and Mr. Sharadchandra Lohokare, (All Capgemini, India) for their constant support and guidance.

References

[1] Philip S. Noe and Kenneth A. Myers,"A Position Fixing Algorithm for the Low-Cost GPS Receiver", IEEE Transactions on Aerospace and Electronic Systems,Vol. AES-12, Issue: 2

[2] E.M. Lassiter,"Navstar Global Positioning System: A Satellite Based Microwave Navigation System", IEEE-MTT-S International Microwave Symposium, 1975

[3] J. Spilker James, "Global Positioning System: Signal Structure and Performance Characteristics", 1978

[4] E. Hammerle , P. Winton and S. Fett,"A multipath-mitigating indoor carrier-phase position tracking system", IEEE Proceedings, Position Location and Navigation Symposium, 2004. PLANS 2004

[5] Frédéric Evennou and François Marx, "Advanced Integration of WiFi and Inertial Navigation Systems for Indoor Mobile Positioning", EURASIP Journal on Advances in Signal Processing, Published 2006

[6] J.I. Meguro, T. Murata, J.I. Takiguchi and Y. Amano, "GPS Multipath Mitigation for Urban Area Using Omnidirectional Infrared Camera", IEEE Transactions on Intelligent Transportation Systems (Volume:10 , Issue: 1)

[7] K. Murakami ,T. Hasegawa , K. Shigematsu and F. Sueyasu, "Position tracking system of everyday objects in an everyday environment", IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010

[8] S. Tang, Hikaridai, Seika-cho, Soraku-gun, N. Kubo and M. Ohashi, "Cooperative relative positioning for intelligent transportation system", 12th International Conference on ITS Telecommunications (ITST), 2012

[9] M. Fujii, R. Ogawara, H. Hatano and Y. Watanabe, "A study on position tracking system for pedestrian navigation using location and sensor information", 13th International Conference on ITS Telecommunications (ITST), 2013

[10] M66 OpenCPU Development Tool Kit, http://www.quectel.com/qdownload/M66_OpenCPU_SDK.html

[11] Eclipse C/C++ Development, https://eclipse.org/cdt/

[12] MentorGraphics Sourcery CodeBench Overview, https://www.mentor.com/embedded-software/sourcery-tools/sourcery-codebench/overview

[13] NMEA Sentence information, http://aprs.gids.nl/nmea/

[14] SMTP protocol standard, https://tools.ietf.org/html/rfc1869

[15] SMTP protocol steps, http://www.facweb.iitkgp .ernet.in/~agupta/IWT/ SMTP-POP3_4.pdf