

GPS-GSM INTEGRATION ON APPLICATION PROGRAMMING INTERFACE

By

Kiran S. Bhat, Sameet M. Deshpande, Sunil A. Chomal



Accord Software and Systems Private Limited,

#37, Krishna Reddy Colony, Domlur Layout
Bangalore - 560071, INDIA
Phone: +91-80-5350105/36/38
Fax: +91-80-5352723.
Email: contactus@accord-soft.com

Biography

Kiran Shivarama Bhat is a Senior Project Leader at Accord Software and Systems. He has been working in the development of GPS core engine and GPS based applications since 1996. He has a Bachelor of Engineering degree in Electronics and Communications from the Bangalore University.

Sameet Deshpande is a Systems Engineer at Accord Software and Systems. He has been working in the software development of GPS receivers since 1999. He has a Bachelor of Engineering degree in Electronics and Communications from the Karnataka University.

Sunil Chomal is a Systems Engineer at Accord Software and Systems. He has been working in the software development of GPS receivers since 2000. He has a Bachelor of Engineering degree in Electronics and Communications from the Calicut University.

Abstract

With the phenomenal growth of GPS based applications, the capabilities of a GPS receiver are expected to be exploited beyond that of a stand-alone receiver. The provision for integrating user applications on GPS chipset is becoming increasingly common among the GPS receivers. Most of the GPS receivers have a dedicated CPU for the application.

This paper describes a distinct application-programming interface (API) on the NAV2300R GPS receiver developed by Accord Software and Systems Pvt. Ltd., Bangalore, India in partnership with Analog devices. NAV2300R is a low-cost GPS receiver developed on a fixed-point DSP platform for automobile applications. The GPS extends the spare memory and processing capabilities on the DSP to the user application through glue less interface. This was successfully used at Accord for the development of a GSM based vehicle-tracking solution for Fleet Management.

1 Introduction

The importance of GPS based applications on day-to-day life has motivated the designers to develop receivers that go beyond simple positioning. Most of the GPS receivers offer spare processing power and resources for integrating user applications. Prominent among them are the hand-held and vehicle tracking applications.

The power consumption, cost, speed and size benefits of fixed-point DSPs make them an obvious choice for high-volume, low-power embedded signal processing applications including GPS receivers and cellular phones.

Accord is an R&D house for developing core GPS engine and GPS based applications. Accord has produced reference designs on both fixed and floating point DSP platforms.

The NAV2300R is a GPS receiver designed around the ADSP2189M fixed point DSP from Analog devices. This receiver offers spare memory and MIPS on the DSP for customization. Accord has developed a GSM based vehicle tracking solution, embedded with the GPS on the same DSP. This was possible by availing the resources of the GPS chipset by using the API on the NAV2300R.

An API is a formalized set of data structures, software calls and routines that can be referenced by an application program in order to access the resources of the supporting system. The API enables the user to embed and execute applications along with the GPS core engine.

2 NAV2300R API

2.1 System Architecture

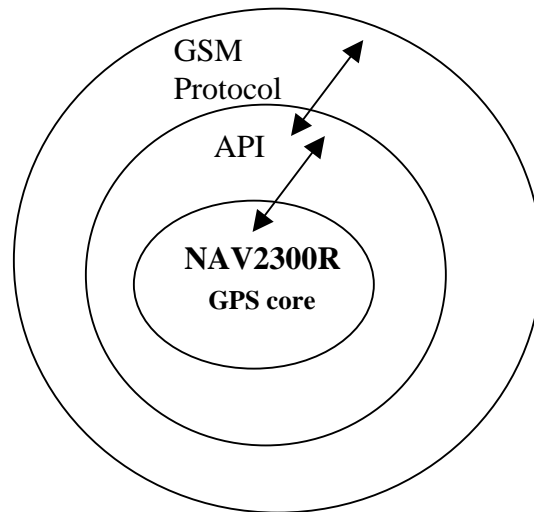


Figure 1: GPS-GSM Application Interface Architecture

Figure 1 shows the architecture of GPS-GSM interface through API. The API allows GSM application to access the resources of GPS core through function calls and shared data structures.

2.2 MIPS and Invocation

The user application is assured of 20 MIPS or equivalently 16 Mega processor cycles per second. The application is invoked once in every 500 milliseconds by the API as shown in Figure 2. This is excluding the brief periods for which the GPS core captures the program control during the execution of the application.

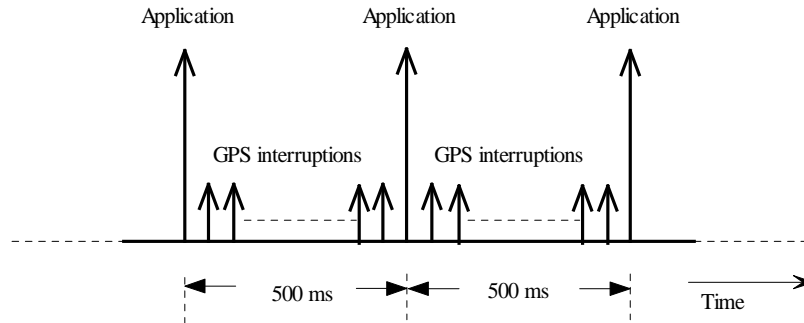


Figure 2: Invocation of GPS & application.

2.3 DSP Program Memory

The API offers 8K words of internal program memory on the DSP for the application. The allocation of program memory is multiplexed during run-time between GPS and application.

Even though the program memory allocated on the DSP is only 8K words, the total size of the application can be up to 20K words. The API expects the application developer to split the application code into modules of size not exceeding 8K words.

2.4 DSP Data Memory

The API provides 8K words of 16-bit internal data memory on the DSP for the application. Unlike the program memory, this is a global memory reserved for the application and is not shared by the GPS core.

Applications requiring power up initializations of the global memory can make use of the 1K words distinguished for this purpose out of the available 8K words. The API manages the initialization of global data memory prior to the invoking of application. In case the application does not require any initializations, this memory segment can be used as regular global memory.

Apart from the 8K words of global memory, the application can utilize 1.5K words of local memory (run time stack) provided by the API.

2.5 UART

The User application can communicate with the external world through the UART of the GPS receiver. The API provides Transmit and Receive buffers for the application to store the transmitted and received message bytes.

The API concerning UART takes care of the actual transmission process and only expects the application to generate the data for transmission. Similarly the API handles the reception process and transfers the data to the application.

2.6 Math Library

An extensive set of Floating-point library routines including arithmetic, trigonometric and matrix operations have been developed specifically for achieving the representation requirements of GPS receiver. The application can make use of floating point data structures using this library.

2.7 GPS Outputs

The navigation solution computed by the GPS core is available to the application through shared data structures. This includes user position, speed and heading, UTC time, parameters of the reference clock such as drift and bias, and raw measurements.

2.8 Software Upgrade

The GPS and Application can be upgraded independently of each other through a serial link using the API.

2.9 EEPROM

A serial EEPROM (Non-Volatile memory) is made use of in the GPS receiver for storing receiver factory defaults and Navigation data. One complete segment on the EEPROM is reserved for the User application.

The application can access EEPROM by requesting the API for read/write operation. API services these requests and indicates on completion to the application.

2.10 Flash Memory

The receiver chipset has an on-chip flash memory of 4 Mega bits capacity. As much as 2 Mega bits of the memory is reserved for storing User application programs and data. The application can request read/write/erase operations of Flash memory through similar requests as in the case of EEPROM.

2.11 Discrete I/Os

The API offers general purpose input and output discretes for the application.

3 *Integration of applications*

Integration of applications require, in general, linking of their object codes to generate a unified executable.

A unique method of integration has been developed, whereby applications can be integrated at the binary level.

In this method, the executable of each application is generated in the raw binary format. These binary files are merged by a utility provided as a part of the integration kit to form the complete solution. This renders the integration process

easy and hassle free. This method of integration allows the applications to be developed independent of each other.

4 *GPS-GSM Embedded Solution*

A vehicle tracking solution has been developed at Accord for Fleet Management application. A Fleet Management System comprises of a control station that gathers location information from a fleet of vehicles each fitted with in-vehicle equipment. The in-vehicle equipment comprises of Accord's NAV2300R GPS Receiver connected to a GSM terminal for updating the location information of the vehicle to the control station.

The interface to the GSM terminal is through the ETSI 07.05 and 07.07 protocols. These protocols are developed as part of the application to be embedded on the GPS chipset using the API.

The application programs stored in the Flash memory are dynamically overlaid on to the internal program memory by the API during execution. The reserved on-chip data memory is made use of for the data structures. The Flash memory is also used for recording of positional information. The EEPROM on the chipset is used for storing vehicle configuration data. The communication with the GSM modem is through the UART resources provided by API.

The application is compiled, linked and split to obtain the binary file. The binary file of the application is integrated to the GPS firmware by the utility.

5 *Advantages of GPS-GSM solution using API*

5.1 No additional CPU

Having the application implemented on the GPS chipset itself, eliminated the need for an additional micro controller.

5.2 DSP core processing capability extended

The application could utilize the specialized instruction set of the DSP core, lacked by the general purpose CPUs. The API provides the application a minimum of 16 Mega processor cycles every second.

5.3 Avail of peripheral resources

Apart from the computational capabilities of the DSP, the application was able to utilize the extensive peripheral resources of the DSP such as the serial ports, interrupts and general purpose I/Os.

5.4 On-chip Memory

DSPs outweigh CPUs and micro controllers in terms of on-chip RAM. Sufficient on-chip data memory was reserved by the API, doing away with additional memory.

5.5 Minimized development time and effort

The additional hardware design turn around time was eliminated. The burden on application development was significantly reduced as the API managed all the DSP system resources.

5.6 Easy Access to GPS information

The application was able to directly access the detailed navigation solution through shared data structures without the need of implementing GPS messages and their extraction.

5.7 Peripherals on GPS chipset

The application could avail the hardware resources such as FLASH and EEPROM interfaced with the DSP using the API. The API managed the actual interactions with the hardware resources.

5.8 Cost

The overall reduction in the development time, effort and the elimination of additional hardware has reduced the cost of developing the application.

6 Conclusion

The API has proven to be very effective in developing GPS-GSM based vehicle-tracking solution. In particular, the versatility of the API and simple integration procedure has contributed to the speedy development of the application.

With the present-day fixed point DSPs, providing enhanced MIPS and memory resources it would be possible to accommodate a wide range of applications on the DSP platform.

Acknowledgement

The authors thank the members of the GPS team at Accord for their contribution during the development of the API and the application development teams for their valuable feedback.