Design and Development of a Data Interface Unit for 
Navigation System of an Automobile

Z. Tracy Austina1, Ashok.C.Meti2, Raveendra Moodithaya3
1-Student, M.Sc. (Engg.), Mechatronics Centre, 2-Professor and Dean – Automotive Engineering Centre, 
3-Professor-VSDC, M. S. Ramaiah School of Advanced Studies, Bangalore

Abstract

Inertial Navigation System blended with Global Positioning System, has gained significance due to enhanced navigation and inertial reference performance. The INS individually can calculate the position of the vehicle without any help from the outside world. However, a large number of errors are introduced by sensors leading to an unacceptable drift in the output. Hence a GPS is used to aid the INS. GPS signals also get blocked under certain circumstances. For best possible estimate of the position of the vehicle, both INS and GPS are supposed to be integrated.

This thesis deals with the design and implementation of a data interface unit for a navigation system of an automobile. In this project, a data interface unit has been developed using PIC18F458 Microcontroller to combine INS and GPS. The developed control algorithm takes the co-ordinates from the Atmel chipset GPS Receiver, if the vehicle is in the range of availability of GPS signals. If the GPS signals are blocked (eg: Inside a tunnel) the Ax and Ay which are very small distances inside the area of non-availability of the GPS signals are calculated and updated with the previous co-ordinates. The accelerometer ADXL202 and Gyroscope ADXR5150 are used to calculate the Ax and Ay. The updated co-ordinates are displayed on the display panel.

But this system has cumulative errors. The INS has a problem of unbounded error and the GPS signals have a bounded error problem. A complete error analysis has to be done and using Kalman Filters or Schuler Tuning methods these cumulative errors can be overcome and the best possible estimate of the position of the vehicle can be obtained.

Key Words: Navigation system, GPS, INS, GPS/INS integration

1. INTRODUCTION

Finding the right path from one place to the other is Navigation. It has become a part of mankind since ages. Right from bees to almost anything on this earth must be able to navigate from one part to the other. People always used to make use of sun and stars for navigation. Over the years we have been able to develop better and more accurate sensors to compensate for our limited range of senses. For any automatic machine, be it a robot, aircraft or other autonomous vehicles, navigation is of paramount importance. There are various means using which navigation can be accomplished, viz. Inertial Navigation Systems (INS), Global Positioning Systems (GPS), dead reckoning systems, radio navigation systems, Doppler heading reference systems, to name a few. Most of the automobiles make use of INS and GPS for navigation.

Inertial navigation affords a method of generating autonomous estimates of vehicle’s position, velocity, attitude and heading based solely on measurements derived from inertial sensors mounted within the vehicle. For the purposes of navigation, it is required to determine position in a defined reference frame, the local geographic frame defined by the directions of true north and the local gravity vector for example. If this is to be achieved it is necessary to provide continuous information regarding the orientation of the accelerometers with respect to the reference frame. The accelerometer output signals may then be resolved in the navigation reference frame prior to their integration. Both GPS and INS have got its own unique characteristics and limitations. INS has the problem of unbounded error in position and velocity estimation. GPS has the problem of bounded error but the position estimate is noisier when compared to INS due to the possibility of blocking of the signal. The GPS has a low update rate when compared to INS. While INS can offer 100 estimates every second, GPS will only output 1 position estimate every second.

Therefore the integration of two systems offers a number of advantages and overcomes each system’s inadequacies. The combination of both technologies is capable of providing more complete spatial data (origin, route, destination) when tracking individual routes.

The motivation of this project is to combine both the systems to overcome their respective inconveniences and hence the best possible estimate of the position and velocity of the vehicle can be done.

2. LITERATURE SURVEY

Across the globe there are thousands who have been constantly working in the field of navigation. They have been trying various possibilities of integrating the GPS and INS through various technologies. Almost all the technologies have their own pros and cons. [1] The principles of the technique are based on the laws of mechanics formulated by Sir Isaac Newton, namely that a vehicle continues to move with the same velocity unless acted upon by an external force and that such a force will produce a proportional acceleration of the body. Hence, all that is required is a continuous measure of the total force vector acting on the body. Given a measure of acceleration, it is possible to perform successive mathematical integrations of the measurement to provide estimates of velocity and position. For the purposes of acceleration, it is required to determine position in a defined reference frame, the local geographic frame defined by the directions of true
north and the local gravity vector for example. If this is to be achieved it is necessary to provide continuous information regarding the orientation of the accelerometers with respect to the reference frame. The accelerometer output signals may then be resolved in the navigation reference frame prior to their integration.

A strap-down system is a major hardware simplification of the old gimbaled systems. The accelerometers and gyro are mounted in body coordinates and are not mechanically moved. Instead, a software solution is used to keep track of the orientation of the IMU (and vehicle) and rotate the measurements from the body frame to the navigational frame. This method overcomes the problems encountered with the gimbaled system, and most importantly reduces the size, cost, power consumption, and complexity of the system.

GPS and INS integrated systems are expected to become more widespread as a result of low cost inertial sensors. The performance obtained from such sensors is generally less than the performance typically associated with tactical-grade inertial sensors. However, with high-quality integration algorithms, it is possible to obtain significant positioning error improvements over GPS-only systems when there is limited satellite availability. [5]

GPS provides the opportunity to record a set of geographic coordinates that specify a particular field location. A way to convert GPS receiver output into linear units for georeferenced data analysis has been illustrated. The GPS output which is an ASCII output is a set of comma-delimited lines (sentences) defined by National Marine Electronic Association interface standard NMEA-0183. It is necessary to record at least one of three sentences: $GPGGA, $GPGLL, or $GPRMC. Each of them indicates geographical position, UTC time (Universal Time Coordinate). The primary tasks of GPS data analysis include determination of the distance, travel speed and heading based on the coordinates of two points. Distance calculation can also be used to establish a local system of coordinates. [6] This project work has been carried out by referring to the references from [7] to [17].

3. HARDWARE ARCHITECTURE OF THE SYSTEM

The typical hardware architecture of the system under development is as shown in Fig 1.

Fig-1 Block Diagram of the data interface unit

The PIC18F458 is a powerful 10 MIPS (100 nanosecond instruction execution) easy-to-program (only 77 single word instructions) CMOS Flash-based 8-bit microcontroller packs Microchip's powerful PIC® microcontroller architecture with a Controller Area Network (CAN 2.0B) peripheral interface into a 40-pin package and is upwards compatible with the PIC16C5X, PIC12C5XX, PIC16C6X, and PIC17CXX devices, providing a seamless migration path of software code to higher levels of hardware integration. The PIC18F458 features a C compiler-friendly development environment, 1024 bytes of EEPROM, self-programming, an LCD, 2 capture/compare/PWM functions, 12 channels of 10-bit Analog-to-Digital (A/D) converter, 2 comparators, the synchronous serial port can be configured as either 3-wire Serial Peripheral Interface (SPI™) or the 2-wire.

Inter-Integrated Circuit (I²C™) bus and Addressable Universal Asynchronous Receiver Transmitter (AUSART). The accelerometer ADXL202 measures acceleration in two axes and has a range of ±2g. Accelerometer is a sensor that measures the acceleration of the body to which it is connected or the devices that measure minute changes in the rate at which they are moving. Here acceleration is the time rate of change of velocity (a=dv/dt). This acceleration is measured in terms of ft/sec or m/sec² or in terms of 'g', which is equal to earth's gravity at sea level (g = 9.81 m/s² = 32 ft/s²). The output signals of the accelerometer are PWM modulated square waves, one for each axis. For 0g the nominal duty cycle is 50%. For 1g in either direction the duty cycle changes by a nominal ±12.5% (37.5% - 62.5% duty). These two signals are connected to the input capture pins of the PIC18F458. The duty cycle of each signal can then be calculated by using interrupts and the PIC's internal 16-bit timer. There are two ways of acquiring the signal from the Accelerometer. One way is to capture the rising and falling edges of the PWM through the XOUT and YOUT pins and the other way is to take the analog data from the accelerometer's XFILT and YFILT pins

The gyrosensor used in this project is ADXRS150, a product of Analog Devices. It can sense YAW rate of 150 degree/sec and with a sensitivity of 12.5mv/degree/sec.

4. SOFTWARE ARCHITECTURE OF THE SYSTEM

The software architecture is a very important phase in the development of a system. This section of the document explains the IDE and the compiler that is been used as well as the complete software architecture of the system. The software that was used to develop this system was MPLAB that is a free integrated development environment for the PIC Microcontroller. It has a source code editor, simulator and numerous other features. Only the simulator was used and the make command for compiling the C source code. The simulations, which are required to be performed on the PIC, are too complicated to be done easily in assembly and so the software was written in C. A company called HTSoft, makes an ANSI C compiler High-Tech
5. RESULTS AND DISCUSSION

The accelerometer Results are as follows:

For the condition of the accelerometer and hardware being level and not moving, the values of \(Dx\) and \(Dy\) (duty cycles) were displayed on the LCD and found to be as follows.

- \(Dx = 51.80 - 52.00\%\)
- \(Dy = 50.25 - 50.45\%\)

With \(0x15\) a specific byte at address \(0x15\) in the PIC's ram is assigned to the variable \(CCP1\). This specific byte is a special function register that holds the first byte of a captured timer value. In the second example bank2 is a reserved word that tells the compiler to place the variable \(Tlx\) in bank2 of the PIC's ram. When using the make command in MPLAB, it compiles the source code by executing the DOS command PICC (High-Tech C's command line driver) followed by various arguments. The arguments used were as follows.

- FAKELOCAL -G -O -D24 -ASMLIST -16F876 -ICD -IC:HT-PIC\_INCLUDE

Different modules like main.c, GPS.c, accel.c, gyro.c and led.c were written individually and later they were integrated.

C for the PIC microprocessors and this is supported by the MPLAB IDE as a third party tool. Thus, with C all the calculations could be done in floating point representation. High-Tech C has some extensions to the ANSI standard that allow it to work easily with the PIC. A couple of examples are as follows.

- Volatile unsigned int captured_CCPI_time @0x15;
- Volatile bank2 double Tlx;

The program will first check whether the GPS signal is available. If it is available then it will display the data from the received. If the signal is not available then the data is displayed from the accelerometer and gyroscopes as explained in flow chart in Fig-2.

### Table 1 - Yaw Rate

<table>
<thead>
<tr>
<th>S no</th>
<th>Angle (Deg)</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>No change</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>Clockwise (R)</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>Anticlockwise (L)</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>Clockwise (R)</td>
</tr>
</tbody>
</table>

The yaw rate results were obtained using ADXRS150 are shown in the table-1 below.

The GPRMC format received from the GPS Receiver is as follows:

SGPRMC,123519,A,4807.038,N,01131.000,E,22.4,084.4,071281,003.1,W*43

RMC Recommended Minimum sentence

123519 Fix taken at 12:35:19 UTC

A Status A=active /* the data status. [A] For valid data, [V] for invalid.

4807.038, N Latitude 48 deg 07.038' N

01131.000, E Longitude 11 deg 31.000' E

022.4 Speed over the ground in knots

084.4 Track angle in degrees True
6. CONCLUSION

The chosen problem was to design and implement of a data interface unit for navigation of an automobile that is based on integrating Inertial Measurements Unit (IMU) and Global Positioning System (GPS) to provide the best possible estimate of the position and the velocity of the vehicle. When GPS and INS are used independently, they have their own disadvantages. INS has the problem of unbounded error in position and velocity estimation. GPS has the problem of bounded error but the position estimate is noisier when compared to INS due to the possibility of blocking of the signal. The motivation of this project was to combine both the systems to overcome their respective inconveniences and hence the best possible estimate of the position and velocity of the vehicle can be obtained.

The estimates were obtained from Accelerometer and Gyroscope. The Accelerometer output work in principle, but was very inaccurate. This is because the accelerations that need to be measured are of the same size or smaller then the noise produced by the accelerometer. Thus the accelerometer is not able to accurately measure the small acceleration that would be produced by a slow moving vehicle. For the system to work accurately then a more sensitive accelerometer has to be used. There are lot of bounded errors in this system that has to be rectified by using Kalman Filters and Schuler tuning methods. A magnetic compass has to be connected to the microcontroller and convert the existing values to the Magnetic reference frame.

REFERENCES
[15] www.htsoft.com (Browsed on 14/08/06)
[16] www.microchip.com (Browsed on 14/08/06)