International Journal of Engineering and Applied Sciences (IJEAS)
ISSN: 2394-3661, Volume-7, Issue-7, July 2020

Implementation and Application of CAN Bus Technology in Multi-source Integrated Navigation System

Cong Wang, Jie Li, Shuai Chen

Abstract—**The integrated navigation system adopting RS-232, RS-422, and RS-485 as the communication bus has the disadvantages of one master and many slaves, data communication as command response, low transmission rate, and weak error handling ability. It cannot meet the requirements of multi-source information fusion. Technical requirements of modern integrated navigation interface. CAN communication bus technology has the characteristics of fast response, high positioning accuracy, good scalability, strong error correction ability, etc., which can improve the flexibility and efficiency of the integrated navigation system. Based on the multi-source integrated navigation system, a plug-and-play interface module based on CAN bus technology is designed, and the driver of the module is written, and the conversion scheme of related sensors in the multi-source system is provided. System debugging shows that the module is stable and reliable, easy to modify and transplant, and can be used in a variety of multi-source combined navigation systems, with high application value.**

***Index Terms*—Integrated navigation, CAN bus, plug and play, positioning accuracy.**

I. Introduction

At present, the existing navigation can only meet the navigation requirements of the near-Earth space, but it cannot provide accurate PNT services in places where the environment is harsh, such as deep sea, underground, indoor, and strong magnetic interference [1].

Compared with the original navigation using information from a single sensor, the combined navigation using multiple sensors can effectively improve the tracking accuracy and state estimation of the downloading body in a complex environment [2-3]. The United States proposed the ASPN plan as early as a few years ago. According to different application task environments, a variety of existing or novel navigation sensors can be flexibly and quickly fused. In my country, with the continuous development of the Beidou system, more and more units and scholars have begun to conduct research on multi-source navigation [4].

With the continuous development of multi-source navigation, it is urgent to re-select the appropriate communication bus technology. Because any node on the CAN network can act as the master node to actively exchange data with other nodes and the information frame of the CAN network node can be prioritized, which provides convenience for real-time control. In addition to the CAN physical The layer and data link layer have unique design technologies, which greatly improve their performance in anti-interference and error detection [5]. Because of this, it is very meaningful to apply the CAN bus to the multi-source navigation system. This paper designed a plug-and-play module based on CAN bus communication and successfully applied it to the DSP+FPGA multi-source combined navigation system [6].

Cong Wang, Collaborative Innovation Center of Advanced Launch System, Nanjing University of Science and Technology, Nanjing, China.

Jie Li, School of Automation, Nanjing University of Science and Technology, Nanjing, China.

Shuai Chen, School of Automation, Nanjing University of Science and Technology, Nanjing, China.

II. CAN Bus Communication Principle

CAN communication protocol mainly describes the way of information transfer between devices. When a node on the CAN bus needs to send a message to other nodes, it will be broadcast to all nodes on the bus. When a node needs to send data to another node, the node needs to send the data and identification The symbol is passed to its own control interface; when the right to use the bus is obtained, the data and the identifier are assembled into a message and the other nodes are in the receiving state. The most commonly used CAN bus is the twisted pair. The signal is transmitted using differential voltage as shown in Figure 1. The two signal lines are called CAN_H and CAN_L. When static, they are both about 2.5 V. At this time, the state is expressed as invisible; use CAN_H ratio CAN_L high means dominant, at this time the voltage values are CAN_H=3.5 V and CAN_L=1.5 V [7].

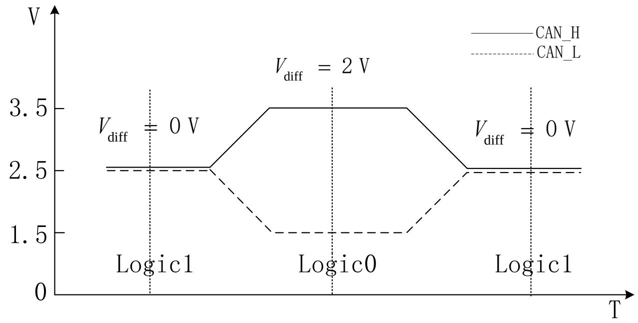

Figure 1 Twisted-pair CAN bus level nominal value

III. System Composition And Working Principle

The overall block diagram of the system is shown in Figure 2. In the system, in addition to MIMU, data is transmitted through the serial port of the GNSS navigation board. Other sensors are used to transmit data through the designed CAN interface module. The system finally outputs navigation by the GNSS navigation board The information is monitored by the host computer. The GNSS navigation board uses the hardware architecture of DSP+FPGA.

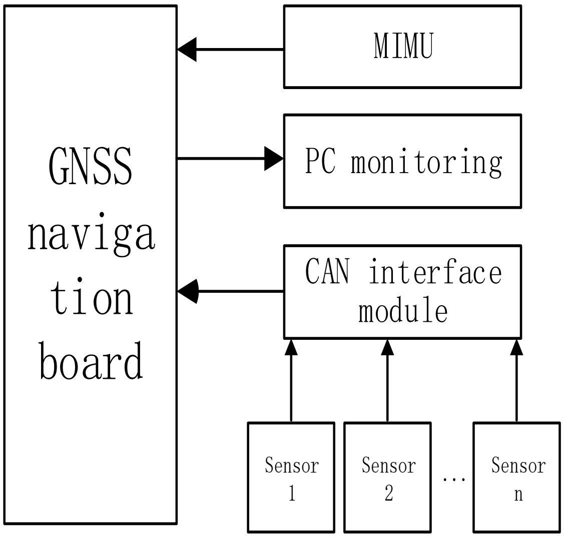

Figure 2 System block diagram

IV. Plug And Play Module Hardware Design

The multi-source integrated navigation system uses a DSP+FPGA architecture. FPGA is mainly used as the baseband signal digital processing module of the satellite navigation system, and DSP is used as the information fusion processing unit of the entire integrated navigation system.

Because DSP selects TMS320C6747, this chip does not have a built-in CAN controller, and it needs an additional CAN controller and CAN receiver to ensure the normal communication between DSP and CAN bus. The CAN controller selects the MCP2515 from Microchip. The chip supports the CAN V2.0B technical specification. The communication rate can be up to 1Mbps, and the SPI interface clock frequency can be up to 10 MHz. It can meet an SPI host interface to expand multiple CAN bus interfaces. The need [8]., and the built-in filter can also reduce the MCU overhead for data processing. The CAN transceiver uses TJA1050 to enhance the differential transmission capability to the bus and the differential reception capability to the CAN controller. In order to further enhance the anti-interference ability, often set up a photoelectric isolation circuit between the CAN controller and the transceiver. Connect the TXCAN and RXCAN pins of the MCP2515 to the TXD and RXD of the TJA1050 through the high-speed photocoupler 6N137, and the two power supplies VCC and VDD used in the optocoupler isolation circuit must also be completely isolated, otherwise the optocoupler isolation will be lost Significance, power isolation can use low-power power isolation module B0505T-1W [9].

Because Weidao's baseband signal digital processing module occupies the SPI communication interface of the TMS320C6747, the SPI and interrupt pins of the MCP2515 are connected to the vacant I/O ports of the DSP to simulate the SPI communication through the I/O to realize the DSP CAN controller communication. Because the recommended voltage value of the GPIO port of the TMS320C6747 is 3.3V and the output of the MCP2515 is 5V, it is necessary to add a 74LVC245 conversion module to the MCP2515 DSP connection. The CAN bus interface hardware structure is shown in Figure 3.

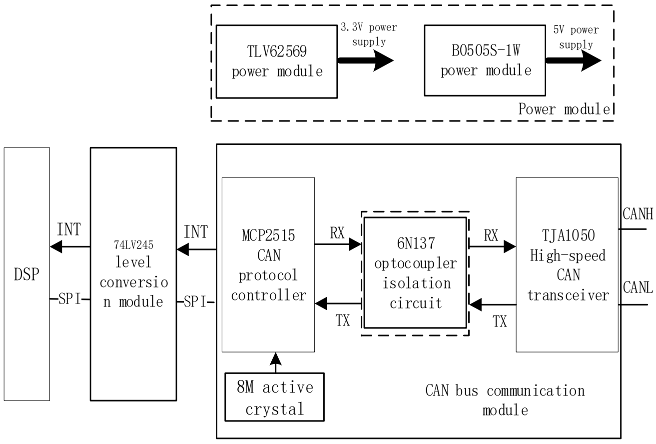

Figure 3 Hardware structure of plug and play module

V. Plug And Play Module Software Design

A. *GPIO Emulates SPI Communication*

First, configure the TMS320C6747 PINMUX register, configure the corresponding pins of the TMS320C6747 as GPIO, then configure the GPIO_DIR register, and configure the corresponding pins as input or output as required to complete the GPIO initialization.

According to the timing diagram, you can write two IO analog SPI timing bit-by-bit read and write functions void SPI_SendByte(unsigned char dt) and unsigned char SPI_ReadByte(void). When the chip select is pulled low, each rising edge of the clock signal reads a Bit value, SPI output timing is the same as this timing, except that reading into becomes sending, that is, reading and writing corresponding data [10]. Then write two DSP functions that write or read values to the register of MCP2515 through SPI of I/O simulation. MCP2515_WriteByte(unsigned char addr, unsigned char dat) and unsigned char MCP2515_ ReadByte(unsigned char addr) One parameter is the address of the register, and the second parameter is the specific value. When reading, the return value is the read value, and the parameter is the address to be read. When writing the read function unsigned char MISO_get(), because it is read bit by bit when reading, it needs to be bit operation to save the entire 8-bit value in an unsigned integer variable.

B. *Software driver design*

Use the interrupt mode provided by MCP2515 to receive and send CAN bus data. The entire interface program is mainly composed of a timer interrupt and an external interrupt [11]. Use timer interrupt to let DSP send data transmission request command and data transmission to MCP2515. The TMS320C6747 timer interrupt trigger can be realized by configuring TGCR, TIM, PRD, INTCLSTAT, TRC and other related registers. When the DSP sends data, first open the SPICLK clock to send the write command, and then send the data to the MCP2515 receiving pin through the SPISMO pin ; When sending, first quickly read some status commands, wait for the TXREQ flag to be cleared, then determine whether the transmit buffer register is empty, write the data to be sent to the transmit buffer register when empty, and write the length of the data to be sent in this frame Enter the send length register of the send buffer, and finally request to send the message, wait when it is not empty [12]. The external interrupt is mainly used to handle CAN bus errors and data reception. GPIO external falling edge interrupt trigger can be realized by configuring the registers of TMS320C6747 ISTP, ICR, IER, ISR, INTMUX and other registers. When the MCP2515 receives data, it is first provided by INT an interrupt to the DSP. After receiving the interrupt signal, the DSP starts the SPICLK clock, sends a read command, and starts to receive the data sent by the MCP2515 transmit pin. When receiving, determine whether a receive interrupt is received. When the receive interrupt exists, receive the information, first read the length of the data received by the receive buffer, and then put the data received by CAN into the designated buffer to end the reception of an ID Clear the interrupt flag bit (the interrupt flag register must be cleared by the MCU), then the next reception can be performed, when the reception interrupt does not exist, wait for the reception interrupt. The flowchart of sending and receiving interrupt subroutine is shown in Figure 4:

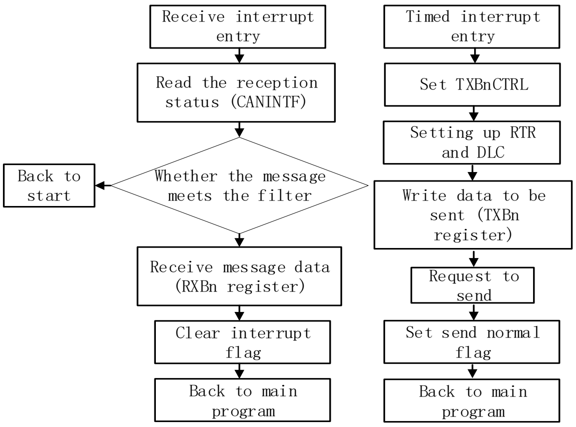

Figure 4 Sending and receiving interrupt program flow chart Hardware structure of plug and play module

VI. Acquisition And Processing Of External Altitude And Speed Information

The altimeter uses the atmospheric pressure sensor model BMP280 produced by Bosch. This sensor measures the temperature and pressure of the outside world. The development board of STM32F103ZET6 receives and processes the data transmitted by the atmospheric pressure sensor and processes and compensates it. The formula for calculating the altimeter is:

$$H = \frac{T_b}{\beta}\left[\left(\frac{P_H}{P_b}\right)^{\frac{-\beta R}{gn}} - 1\right] + H_b \qquad (1)$$

Among them, H is the altitude required, P_H is the atmospheric pressure at that altitude, the air-specific gas constant is $R = 287.05287\mathrm{m}^2/(\mathrm{K}\cdot\mathrm{s}^2)$, the free fall weight force acceleration is $g_n = 9.80665(\mathrm{m/s}^2)$, the vertical rate of temperature change is β, T_b、H_b、P_b are the lower limit of the atmospheric temperature, standard atmospheric pressure altitude and atmospheric pressure of the corresponding layer in the altitude stratification adopted by the international standard atmosphere.

Since the absolute accuracy of BMP280 is not relatively high, in order to ensure that the altimeter can transmit accurate height information, the satellite guide module in the DSP needs to send an initial height information to STM32F103ZET6, because the DSP serial port is sent in TTL format, so SP3232 is required The TTL signal is converted to 232 communication. Since the data transmitted by the altimeter can be subcontracted and processed on STM32, it can be set to fixed byte transmission. Therefore, the UT-2505 module can be used to directly convert the height information output by STM32 into The CAN message is sent to the designed plug-and-play module.

The external speed information is obtained by an odometer, which generates a series of pulses by detecting the magnetic sheet on the vehicle's rotating shaft, Through the pulse count P in the sampling period T_{odo}, the driving distance of the vehicle can be obtained, so that the forward speed $\hat{V}_{odo}$ of the vehicle can be calculated.

$$\hat{V}_{odo} = \frac{K_{odo}(P+w)}{T_{odo}} \qquad (2)$$

w is the measurement error, K_{odo} is the scale of the odometer (ratio of travel distance to pulse number within a period of time). After the second calibration, it can output relatively stable speed information. In this design, the odometer itself can be output in the form of CAN communication, so it can be directly connected to the plug and play module. But the odometer default CAN bus baud rate is 500Kbps, so you need to configure the plug and play module, use the CAN baud rate calculator to calculate and configure the values of the three registers CNF1, CNF2, CNF3 in the MCP2515.

VII. Test Results

In order to detect whether the CAN bus technology is effective in the multi-source navigation system, a dynamic sports car experiment was carried out, in which two sensors were mounted on the multi-source navigation system and chose to conduct it on an outdoor open road section, taking the foreign high-precision receiver SPAN-KVH1750 as a benchmark, SPAN-KVH1750 and a multi-source integrated navigation system with CAN interface The position error and speed are shown in Figure 5 and 6.

Experiments show that the design of the plug-and-play module based on the CAN bus is reasonable, and it provides the function of plugging and unplugging the sensor for the navigation system.

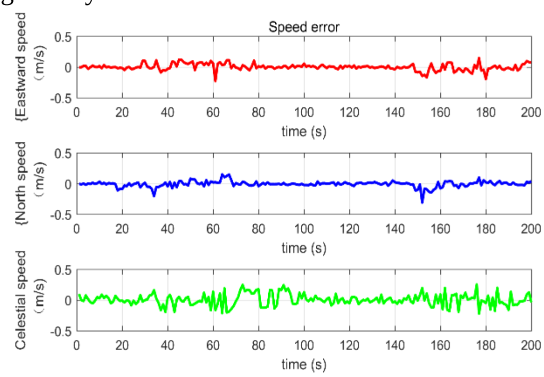

Figure 5 Position error curve

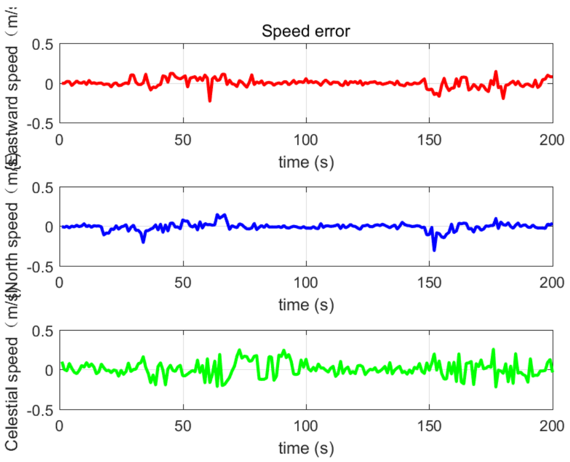

Figure 6 Velocity error curve

It can be seen from the position error curve shown in Figure 10 and the speed error curve shown in Figure 11, latitude Latitude error is 1.23m (1 σ), longitude error is 1.65m (1 σ), and height error is 3.43m (1 σ). The eastward speed error is 0.13m/s (1 σ), Northbound speed error is 0.1m/s (1 σ),and the speed error in the sky is 0.18m (1 σ).

This time, a plug-and-play module that can be used for multi-source combined navigation is designed based on CAN bus technology. This module has the characteristics of reasonable configuration of time resources and space resources, reliable matching of interfaces and timing, easy portability, and various implementation methods. After many tests, the experiment shows that the module has strong anti-interference ability and can be effectively applied to multi-source integrated navigation system. This module has good development prospects in unmanned vehicles, satellites, missiles and other carriers. But there is no relevant test research on other chips, and you can try to change the platform later.

Acknowledgment

Cong Wang thanks the team he belongs to, it is the team that provides a good experiment platform.

References

[1] Song Lijun, Xue Lianli, Dong Yanqin, Wu Yaming, Chen Xiaozhen. Development and suggestions of Quanyuan positioning and navigation [J]. Navigation and control. 2017, 16 (06): 99-105+24
[2] Félix Perosanz. GNSS: A revolution for precise geopositioning [J]. Comptes rendus – Physique. 2019, 20(3):171-175.
[3] Ambrogio Maria Manzino, Paolo Dabove, Neil Gogoi. Assessment of positioning performances in Italy from GPS, BDS and GLONASS constellations [J]. Geodesy and Geodynamics. 2018, 9(06): 439-448.
[4] Dai Haifa, Bian Hongwei, Ma Heng, Wang Rongying. Construction of a unified theoretical framework for all source positioning and navigation [J]. Navigation positioning and timing. 2018, 5 (06):9-16.
[5] Zhou Hongying. Signal timing analysis of CAN bus [J]. Automotive practical technology, 2018 (23):259-26.
[6] Zhao Wanlong, Meng Weixiao, Han Shuai. Overview of multi-source integrated navigation technology [J]. Telemetry and telecontrol. 2016, 37 (06):54-60.
[7] Niu Yueting, Zhou Ligong, Fang Dan. Embedded development of CAN bus [M]. Beijing University of Aeronautics and Astronautics Press, 2016.
[8] Sun long, Liu Huihu. Design and implementation of CAN bus control software based on MCP2515 [J]. Modern information technology. 2018, 2 (07): 1-4+8.
[9] Wang Xiang, Lang Bing. Application of MCP2515 in DSP can port expansion [J]. Shanxi electronic technology. 2007 (06): 8-9+12.
[10] Sun song. Design and implementation of GPIO analog SPI bus master-slave bidirectional communication in embedded system [J]. Automation technology and application. 2007 (10): 38-39+42.
[11] Ding Xuejing, Xu Yonghui. Design of CAN bus communication unit based on MCP2515 [J]. Modern electronic technology. 2015,38 (21): 60-63.
[12] Wang Xiaowei. Design and implementation of CAN bus communication system based on DSP and FPGA [J]. Electronic design engineering. 2018, 26 (21): 67-70 + 76.

Cong Wang a postgraduate of Collaborative Innovation Center of Advanced Launch System, Nanjing University of Science and Technology, Nanjing, China. Research on integrated navigation, satellite navigation.

Jie Li a postgraduate of School of Automation, Nanjing University of Science and Technology, Nanjing, China. Research on integrated navigation, satellite navigation.

Shuai Chen an associate professor of School of Automation, Nanjing University of Science and Technology, Nanjing, China. Research on integrated navigation, inertial navigation.