Design of Heartbeat Monitoring Based on Serial Port and its Implementation in QNX

Xukan Li*

Project construction management department, Zhejiang zheneng Aerospace Hydrogen Energy Technology Co., Ltd, 5th floor, No. 201, Kaixuan Road, Kaixuan street, Shangcheng District, Hangzhou 310020, China.
*Email: lixukan@foxmail.com

Abstract. Today, with the rapid development of information technology, the stability of information systems is the guarantee for the normal operation of business systems and data security. Heartbeat monitoring technology enables high system availability and a high degree of system security at a lower cost. In order to ensure the stability of the information system, the heartbeat monitoring technology has been studied. Firstly, the principle of heartbeat monitoring technology is discussed. The basic heartbeat monitoring method is implemented between the main system and the standby system or any two connected systems through the serial port. Then the QNX is used as the technology to implement the carrier to RS-232-C. Finally, for the related modules such as heartbeat sending, heartbeat receiving and processing, the program working flow chart is presented, some examples are given, and interfaces are reserved for function expansion for reference by relevant engineering and technical personnel. In practical applications, based on the actual situation, emergency procedures and recovery procedures can be designed to improve the emergency response capability and reliability of the overall system.

Keywords. Information system stability; Asynchronous message; Heartbeat monitoring; QNX real-time operating system; Serial communication interface; Timer.

1. Introduction
With the continuous development and popularization of information technology, many core business systems are built on the information system, which puts forward higher requirements for the security and stability of the information system. These business systems include a series of interconnected applications, giving birth to a large number of technical requirements for mutual communication and message passing between applications.

There are multiple ways to send messages to each other between applications, which can be generally divided into two categories: synchronous messaging when sending messages in time synchronization, and will asynchronous messaging when sending messages in time synchronization[1]. Synchronous messaging requires strict matching of the running steps of both the sender and the receiver, which is generally difficult to achieve.

Heartbeat monitoring is a very important method of asynchronous message delivery. In order to keep abreast of the real-time running status of the application, discover occasional abnormal situations in time, and deal with failures in time, heartbeat monitoring technology is playing an increasingly important role in eliminating local failures and improving the reliability of the entire system. Therefore, it is of great significance to study how to realize heartbeat monitoring.
QNX real-time operating system (Real Time Operating System, RTOS) is a very important distributed real-time operating system ("QNX" for short), and the research in this article is mainly based on this system.

2. Heartbeat Monitoring and Serial Port

2.1. Heartbeat Monitoring
Heartbeat monitoring usually refers to sending heartbeat signals at an agreed time interval between any two connected systems and judging the other party's system health by whether the heartbeat signal can be received or not, and the condition of the heartbeat signal received. In the active and standby system environment, when the system in the standby role receives the heartbeat signal and checks it correctly, it means that the system in the main role is running normally; When the system in the standby role does not receive the heartbeat signal for several consecutive times (which can be customized according to the demand), or the heartbeat signal received is wrong, the standby system judges that the main system is abnormal and switches itself to the main system to avoid the interruption of the tasks performed by the failed main system, so as to improve the reliability of the overall system[2].

In order to study and realize the function of heartbeat monitoring, considering the universality of the system, the heartbeat signal can be transmitted through standard communication interfaces such as serial port, and other relevant information can be transmitted at the same time, so as to improve the efficiency of hardware use and reduce the implementation cost of heartbeat monitoring technology.

2.2. Serial Port
Serial port is the serial communication interface, which generally includes RS-232-C, RS-422, RS-485, USB, etc. The RS-232-C standard was proposed the earliest and the most widely used. It is currently the most commonly used serial communication interface [3].

This article takes RS-232-C interface as the communication interface between applications.

3. Principle of Heartbeat Monitoring
Generally speaking, the basic heartbeat monitoring can be realized through the periodic sending, receiving and processing of independent heartbeat signals through the serial port between the application of the main system and the standby system or any two connected systems. Therefore, the realization process of heartbeat monitoring mainly includes two parts: heartbeat signal sending, heartbeat signal receiving and processing. The sender is responsible for heartbeat signal sending, and the receiver is responsible for heartbeat signal receiving and processing. In order to realize the function of heartbeat monitoring, generally the sender is also the receiver.

3.1. Heartbeat Signal Transmission
The sending of heartbeat signal is completed by the sender. How the sender sends heartbeat signal needs to be designed according to the requirements of the overall system, mainly including heartbeat message, heartbeat cycle, abnormalities handling, etc.

Heartbeat messages are text, numbers, or other data types that can be recognized by the system, and multiple sets of heartbeat messages can be defined as needed to achieve different functions. Heartbeat cycle, also known as heartbeat frequency, is used to control the transmission frequency of heartbeat signal, which needs to be determined according to the work performed by the system and the processing speed of the system. If the working time required by the system is high, the heartbeat frequency should be increased; If the processing speed of the system is relatively slow, the heartbeat frequency shall be reduced accordingly, which shall not affect the normal operation of the system. The heartbeat frequency can be controlled by timer [4].

Abnormalities handling refers to the handling measures when the heartbeat signal is sent incorrectly, which can be defined according to the system requirements.
3.2. Heartbeat Signal Reception and Processing
The receiving and processing of the heartbeat signal is completed by the receiver. How the receiver sends the heartbeat signal also needs to be designed according to the requirements of the overall system, including heartbeat messages, heartbeat cycles, and abnormalities handling. The receiver's heartbeat message and heartbeat cycle must be consistent with the sender's.

Abnormal handling refers to the processing measures when the heartbeat signal is not received, the abnormal heartbeat signal is received, or the specific heartbeat signal is received. Generally, when the heartbeat signal is not received or the abnormal heartbeat signal is received, it can be handled according to the other party's system failure; When a specific heartbeat signal is processed, it can be processed in an agreed way to realize the special functions of the system. When receiving heartbeat information, the heartbeat status should be set and relevant data should be recorded.

4. Implementation of QNX and Serial Communication

4.1. Introduction to QNX
QNX is a very widely used RTOS. QNX Momentics can be used to visualize the target system under Windows and other platforms.

The real-time performance of the operating system means that the clock signal can be timed accurately, the clocks everywhere can be consistent, and the QNX can reach the microsecond level in different occasions. The QNX is based on a priority preemptive scheduling strategy, which is different from the Windows equal-time operating system which uses time slice management. Once the high-priority tasks in the QNX are ready, they can immediately occupy CPU resources.

4.2. Construction of Serial Communication Based on QNX
When the QNX is started, it will load a system image file including the operating system, executable programs and any related data files, also serial port is configured at this time.

4.3. Features and Implementation Methods of QNX Serial Communication
QNX access to the hardware is very simple. For users, the operation of the hardware is the operation of the file, and the hardware file is stored in the "/dev" directory of the QNX.

Serial communication requires the speed and message format of both parties to be consistent. The speed parameter is mainly the baud rate, and the message format mainly includes data bits, stop bits, and parity bits. In the QNX, these parameters are included in the "termios.h" header file, which can be set by bit operations (set, clear, etc.).

QNX implements the operation of the serial port by accessing the serial port file. The serial port file naming rule is "ser+number", such as "/dev/ser0".

The serial communication flow chart of QNX is shown as in figure 1.
5. Implementation of Heartbeat Monitoring of QNX

5.1. Heartbeat Monitoring Module
The heartbeat monitoring system based on serial communication of QNX includes two modules: heartbeat sending module and heartbeat receiving and processing module.

5.1.1. Heartbeat Sending Module. The heartbeat sending module continuously writes predefined heartbeat messages to the serial port according to the heartbeat frequency. Accordingly, certain fault diagnosis capabilities are required when sending abnormalities.

The working flowchart of heartbeat sending module is shown in figure 2.

Figure 1. Serial communication flowchart of QNX.

Figure 2. Working flowchart of heartbeat sending module.
5.1.2. *Heartbeat Receiving and Processing Module.* The heartbeat receiving and processing module continuously attempts to receive data. Once the data is received, it is compared with the predefined data. If the results are consistent, it is determined that the heartbeat signal is received; If no data is received beyond the specified time, perform a predetermined task.

The working flowchart of heartbeat receiving and processing module is shown in figure 3.

![Flowchart of Heartbeat Receiving and Processing Module](image)

**Figure 3.** Working flowchart of heartbeat receiving and processing module.

5.2. *Heartbeat Coordination Timer*

In the heartbeat monitoring system of serial communication of QNX, the sending, receiving and processing of heartbeat information need to be coordinated by timers [5], which can be realized by timers provided by QNX. An example is given below.

```c
for (;;) {
  rcvid = MsgReceive(chid, &msg, sizeof(msg), NULL);
  if (rcvid == 0) {
    /* we got a pulse */
    if (msg.pulse.code == MY_PULSE_CODE) {
      printf("Get a pulse signal from the timer \n");
      /* other pulses */
      /* other messages */
    }
  } /* other messages */
}
```

Each time a pulse is received, the heartbeat coordination timer completes a timing to realize the scheduling of heartbeat signal sending, receiving and processing.

5.3. Verification of Heartbeat Monitoring

In order to verify the realization of heartbeat monitoring, a test platform was built, including 2 sets of systems, each of which contains an independent CPU connected with RS-232-C.

By compiling the heartbeat monitor program into the QNX system image files of the two systems, the heartbeat monitor program is started with the QNX. After the heartbeat monitor program is started, the system starts to receive the heartbeat signal from the RS-232-C interface. If the signal is not received, it will be set as the master system, otherwise it will be set as the slave system. After the master-slave system is determined, the master system continuously sends heartbeat signals, and the slave system continuously receives heartbeat signals; when the slave system fails to receive the
heartbeat signals, it automatically switches to the master system and sends a reset signal to the master system.

6. Conclusion

Under current conditions, the dependence of various business systems on information systems is increasing day by day. The cost of heartbeat monitoring technology is low, and it has a better effect in improving the safety and reliability of the overall system.

This article mainly discusses the heartbeat monitoring method and realizes the basic heartbeat monitoring through serial port under QNX. In practical application, relevant engineers and technicians can design emergency measures and recovery procedures on this basis, or use the heartbeat monitoring system to achieve the required specific functions.

References

[1] Zhu Y, Yao H and Xu W 2007 Message transfer mechanism and routing algorithm in message transfer network Journal of Shanghai University 13(5) 611-616
[2] Ni M, Pan Y and Hu Y 2007 Design and implementation of heartbeat monitoring based on JMS asynchronous messages Computer Applications and Software 96-98
[3] Zhou R, Yang F and Chen P 2001 Common serial communication interface standards Electrical Measurement and Instrumentation 33-35
[4] Zhao C, Zhang C and Chen L 2018 Stepper motor control program design based on timer Precision Manufacturing and Automation 30-32.
[5] Liu D and Zhou Y 2004 Embedded system serial communication based on QNX Journal of Zhejiang Institute of Technology 25-29