DOI:10.13656/j.cnki.gxkx.1995.02.004

Multitask Parallel Mode in Process Control

过程控制中多任务并行模式

Ghen Fengmou

陈凤谋

(Comptutation center of Guangxi University for
Nationalities, Xixiangtang, Nanning, Guangxi, 530006)
(广西民族学院计算中心 南宁市西乡塘 530006)

Abstract A real - time operating system (RTOS) running in process control system is introduced. It manages to find time to run other scientific caculation programs on the condition of giving priority to the control tasks.

Key words process control, multitask, parallel mode, operating system, multiple, destination program

摘要 介绍一个过程控制中的实时操作系统（RTOS），在完成过程控制任务的条件下，还可以抽空运行其它的科学计算的程序，所以该系统实际是一个多道程序系统。

关键词 过程控制 多任务 并行模式 操作系统 多道目的程序

The main function of a real - time operating system (RTOS) is as follows:

. Optimal dispatching of the multiple destination program.

. Analysis and process of interrupt.

. Management and allocation of I/O device.

. Extra - instruction analysis and management.

. Information management.

. Man - machine communication.

1 Optimal dispatching of multiple destination program

In process control the multiple destination program is actually the combination of several control tasks.

The characteristic of this RTOS realized not only the multitask concurrent execution in process control, but also other program controls besides control tasks.

1.1 Priority of program

In fact, the multiprograms coexisting in computer make use of CPU serially and segmentally in microcosm. The priority sequence of running is decided by the programmer according to the order of priority of each program. Certain number of levels according to the order of priority is called the priority of program. Each program in computer corresponds to a fixed priority. RTOS divides the programs coexisting in computer into five levels.

The first level is the highest one and the fifth level is the lowest.

The fifth level is for the other scientific caculation programs that have no relationship with the control. This arrangement reflects the thought of ensuring control prior to caculation.

The function of RTOS dispatch program is to select program with highest priority from programs raising running requests and put it into running.

The number of programs in the same priority level can reach sixteen. They are given program orders in sequence. The order is 0, 1, 2,... 15. For example: program 301 refers to the program NO. 1 in the third

1994-04-08 收稿.
1994-08-03 修回.

level. 301 is the name of program and also the reflection of it's priority.

1.2 Program state

Under RTOS control and at certain time，each program coexisting in the computer must be in one of the following states or in a conversion state.

1.2.1 Running state

If CPU is executing the instruction of a program，it means that the program is in running state.

1.2.2 Request state

A program has possessed of the running condition，but the dispatch program hasn't allocated CPU to it bacausc of its low priority，So it is standing in queue for dispatch. Once the progsam gets CPU it will be put into running immediately. We call the program request state.

1.2.3 Suspension state

For certain reason，a program can't continue to excute and has to be paused in running procedure (for example：to I/O)，we call it suspension state. Only when the reason caused suspension disappear and suspension is relieved will it switch to request state.

1.2.4 Closed state

A program in closed state means that it's executive right is cancelled. Only after openning will it regain the executive right.

1.2.5 Exit state

The program will become exit status after it has completed running correctly. The program exited means no longer joining queue.

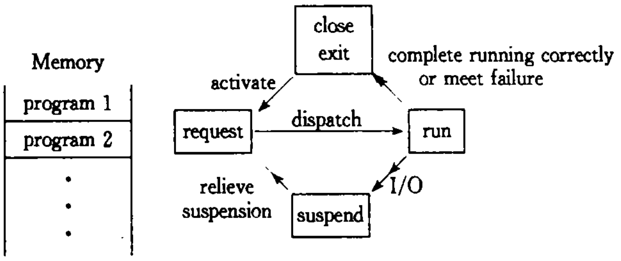
1.3 Program status word

The state of program in computer is identified by corresponding status word. So each program has its request status word，namely excution status，suspension status and closed status words.

1.4 Conversion of program states

Dispatching the multiple destination program is realized by changing program status word continuously according to certain algorithm. The principle of dispatch is：Selecting a program with highest priority level among those having joined queue (in request state) and putting it into running.

STATE CONVERSION SCHEMATIC DIAGRAM



2 Analysis and process of interrupt

2.1 Classification of interrupts

Under RTOS control，there are following species in point of interrupt sources：

- Power－failure interrupt；
- External device interrupt；
- A/D，D/A，etc. interrupt from production field；
- Interrupt caused by man－machine communication on control desk；
- Interrupt caused by using extra－instruction in program (also called voluntary interrupt)

The methods of RTOS to process different kinds of interrupt are different. It processes the interrupt by starting corresponding interrupt process subroutine. So there are：

- Power－failure interrupt process subroutine
- External device interrupt process subroutine
- Keyboard order of man－machine communication interpreter and so on.

2.2 Interrupt queue and priority

2.2.1 Priority of interrupt

The urgent degree of different kinds of interrupt to be processed has nothing in common with each other. So various interrupts are divided into several levels in advance according to the differences of their urgent degree. When several interrupts of different priority appear at the same time，the urgenter one will be responsed with priority and processed timely to ensure the security and reliability of production procedure.

2.2.2 Principles of interrupt queue

- When interrupt request with different degrees of priority happened at the same time，the interrupt request with highest priority will take precedence.
- Once a higher interrupt request happened during processing a interrupt，it will pause processing the

interrupt with lower priority but process the interrupt request with higher priority. When finished, it will return to the interrupt which has just been held up and continue to process. In other words, interrupt permit nest.

. It is not allowed to break the current running interrupt process program if the other interrupt is in the same level as the current one or in lower priority level.

2. 2. 3 Realization of interrupt queue

There are three methods of solving interrupt queue:

. Realized by hard device through special queue line. The more the varieties of interrupts are, the more the deviced are needed.

. Realized by program of RTOS. It means to make use of program test method, the priority sequence of interrupt depends on the sequence of program test. The one tested first will be processed first. So its priority level is higher.

. Combination of software and hardware. On basis of the hardware queue, it forbids some devices to give out interrupt request by setting "interrupt mask" instruction, so as to fullfill the purpose of changing priority sequence.

2. 2. 4 The main interrupt conducts the following tasks:

. Close interrupt;

. Protect information of breakpoint field;

. Test if there is power - failure;

. Search the interrupt resource and switch to corresponding interrupt process subroutine;

. Open interrupt.

2. 3 Analysis and process of extra - instruction

In multitask concurrent mode, there are likely to be two or more programs request certain resource at the same time. To realize resource - sharing and avoid contending for system resource, RTOS offers some commands (called extra - instruction) to advance requests for system resources in the forms of command. The function of these commands are executed in hardware meaning but realized by software interpreter. RTOS offers sixteen commands divided into the following species:

. Command about calling external device;

. Command about using variable relative time;

. Command about calling standard subroutine;

. Command about calling destination program.

2. 3. 1 Form of command

Among them, parameter 1 to 4 are the concerned information that must be offered to realize the current command. For example, a program should offer the starting address in memory and the quality need to print of these informations, if the program starting a printer to print out a batch of information in memory. Here the "starting byte address" and "quantity" is offered in form of parameter.

JSR (main entry)
Command code
Parameter 1
Parameter 2
Parameter 3
Parameter 4
Error return address
Normal return address

2. 3. 2 Main process program of command

CPU will produce software interrupt and switch to RTOS main process program of command if it met the command as above in the procedure of executing a task. The working procedure is as follows:

. Close interrupt;

. Protect the field;

. Check if the command is legal. If it is legal it will switch to corresponding command inpreter entry and realize it.

. If not legal, it will error return and run the error process.

2. 3. 3 Command examples

------- about using variable relative time command

Some work of computer in controlling production procedure need to run at fixed or no - fixed time.

RTOS offers three methods of using time on basis of real - time clock:

1. Absolute time: not only to count for arranging three shifts but also to be time basis producing calendar.

2. Fixed relative time: some work in production procedure need to go on at fixed time, such as travel-

ling test program, timing tabulator program, They will start running when the stipulated time is up. The time separation stipulated in advance is called fixed relative time (of course fixed relative time of each program wouldn't be the same). RTOS scans the fixed time table continuously. It will start corresponding program to complete stipulated function when the stipulated time is up.

3 Variable relative time

The fixed relative time is actually a fixed period. It is stipulated in advance how many programs need to run periodically. The length of period is also stipulated in advance. But some work in production procedure can't be stipulated in advance, they may need to delay time or pause running temperarily so as to coordinate to complete a task together with other programs. The time of delay and pause is not defined. It is called variable relative time. RTOS sets up a program name list using variable time. Corresespondingly there is a variable relative time integrating table. The concerned program can fulfill the purpose of delaying or pausing for a period of time with command using variable relative time according to these two tables.

2.4 Management and allocation for external device

To avoid the multiprogram requesting to use external device at the same time and causing contradiction. RTOS stipulates that all the user programs should only use "command of calling external device" to request from RTOS. RTOS will conduct dynamic allocation according to the priority of the requester and current state of concerned external device after having received request.

2.4.1 State of external device

This means the state whether the external device is serving for a program. It is called service state if it is serving for a program, otherwise it is called idle state. Work state of each external device is marked with two status words.

(1) Device request status word: it signs which level (or which levels) of program is (or are) standing in queue for requesting using the device currently.

(2) Device service status word: it signs that the device is serving for a program currently.

2.4.2 Device allocation principle of RTOS

(1) If there are several levels program request for using the same device at the same time. RTOS will allocate it to the program with highest priority among those joining request queue on the condition that the device is in idle state (service status word is 0).

(2) If a device is serving for a program and there are other programs request for the device, The device will continue to serve the former program, no matter what priority level these later programs are. It will allocate the device to the program with highest priority joining request queue until the service is completed. In other words, once any device is starting work, the other programs can't break it in the half way.

2.4.3 Device interrupt process

Interrupt processes of different devices are different. The device gives out interrupt request so as to report its work state to CPU. Interrupt of I/O device is generally processed as follows:

. Reset device;

. As for input device, send the information in device buffer unit to memory through accumulator;

. Check if the total information is conveyed completely. If complete, check if there is program in relieve suspension state because of waiting for completion of current device's input work. If there is, relieve suspension status word with "0". If not complete, then restart.

. Then check request status word of the current device. If there is program in request queue, then start the device. If there isn't, stop the device.

2.5 About man - machine communication

The operator should give some commands to computer through keyboard to interfere manually for grasping the status of computer controlling production. RTOS sets up several keyboard commands. This is realized by keyboard command interpreter.

(责任编辑：莫鼎新、蒋汉明)