CONTROL SYSTEM FOR THE RIKEN SSC (II)

K. Shimizu, M. Nagase and I. Takeshita
The Institute of Physical and Chemical Research
2-1 Hirosawa, Wako-shi, Saitama 351-01, Japan

ABSTRACT

The RIKEN SSC control system uses new intelligent modules, CIM's and DIM's. A CIM/DIM system which consists of a CIM and several DIM's can communicate with 256 input and output ports of DIM's directly. Furthermore, it has a function as a local controller for a group of apparatus and can include 256 kinds of macro-programs for sequential control, data logging, etc. Principle of operation, instruction format and command function of the CIM/DIM system are described.

INTRODUCTION

Construction of the RIKEN SSC control system is based on an idea described in Ref. 1. This system consists of four layers; a mini computer for a central control, CAMAC equipments such as highway driver and crate controller, communication interface modules (CIM's) mounted in a crate and device interface modules (DIM's) placed in one or several apparatus. The CAMAC equipments are used as media to standardize hardware and soft ware of the communication system between the control computer and each CIM/DIM system. In the control system, the CIM/DIM system is characterized in that it plays a role of local control in addition to an interfacing system. Architectures, functions, advantages and test results of the CIM/DIM system have been described.[2,3]

In the present conference, essentials of the CIM/DIM system are made clearer by explaining principle of operation, instruction formats and command functions.

PRINCIPLE OF OPERATION

The CIM/DIM system is operated by an instruction from a control computer. Figure 1 shows the operation flow diagram of the CIM. When the power is made on, the CIM executes "Initialization Program"; various registers in the CIM are set into each initial value and then a Q-flag is set into logical 1. The Q-flag indicates whether the CIM is "Ready" (Q=1) or not (Q=0). It is used as a response signal of the CAMAC equipments. In the ready situation, the CIM waits for an instruction from the control computer. When an instruction "Task" is found by CIM (yes), CIM decodes and executes it. CIM sends Task to the DIM depending on its kind. After the Task is executed, CIM advances to a next step "DIM Request". If none of Task is found by CIM (no), "Execute" step is skipped.

When DIM executes a macro-program or finds a failure of apparatus connected therewith, it requires a service from the CIM. In the step, the CIM checkes the service request coming from DIM's. If a service is required, CIM sets a LAM-flag and returns to the ready situation. If no service is required, it jumps to the ready. If an interrupt is requested by the control computer during the execution of the macro-program, the CIM interrupts the execution, jumps to the ready situation and waits for a next instruction of the control computer. Thus, CIM responds to the instruction of the control computer at any time. CIM can check a request from the DIM quickly because it has a register for accepting service requests from all the DIM's.

Figure 2 shows the operation flow diagram of DIM. DIM as well as CIM reachs "Ready" after the execution of its "Initialization Program". In "Ready", DIM polls instructions coming from CIM which communicates with DIM. When DIM finds an instruction "Task", the DIM decodes and executes it, and returns to the ready situation. If an interrupt is request from the control computer during the execution of a macro-program, the DIM interrupts the execution and jumps to "Ready". The request is accepted because the interrupt request has a top priority.

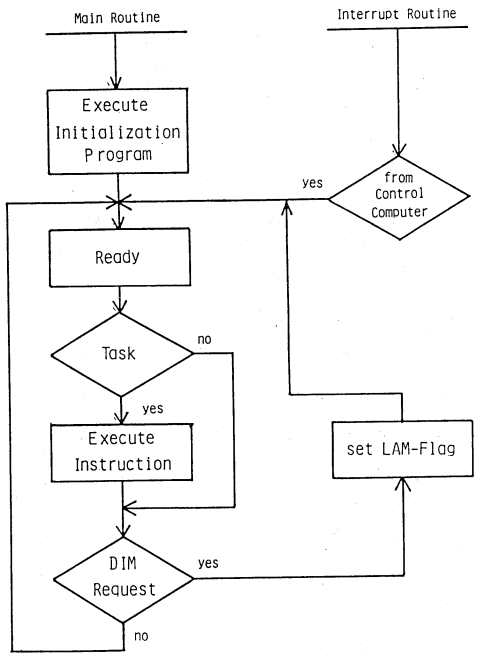
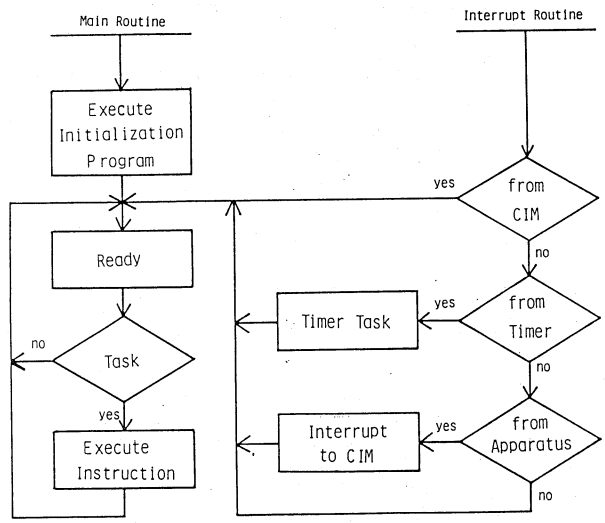


Fig. 1 Operation flow diagram of CIM.

Fig. 2 Operation flow diagram of DIM.



In figure 2, "timer Task" is a routine for fetching the information of all input ports of DIM and storing them in their respective file. DIM has a timer. If the timer task is activated by an instruction of the control computer, the DIM executes a timer task every time when an interrupt is requested from the timer. DIM has a function of detecting an apparatus failure. If an apparatus connected with an input port of a DIM fails, the DIM requires a service, above-mentioned "DIM Request", from the CIM. Priority of the interrupt request "from Apparatus" is lower than that "from Timer". That is, the interrupt request due to an apparatus failure is processed after the end of a timer task.

INSTRUCTION FORMAT

Instruction format for operating the CIM/DIM system consists of three bytes. Higher four bits of the most significant byte are used for specifing a command, lower four bits of the most significant byte, the middle byte and the least significant byte are used as an operand. We have five kinds of instruction formats as below.

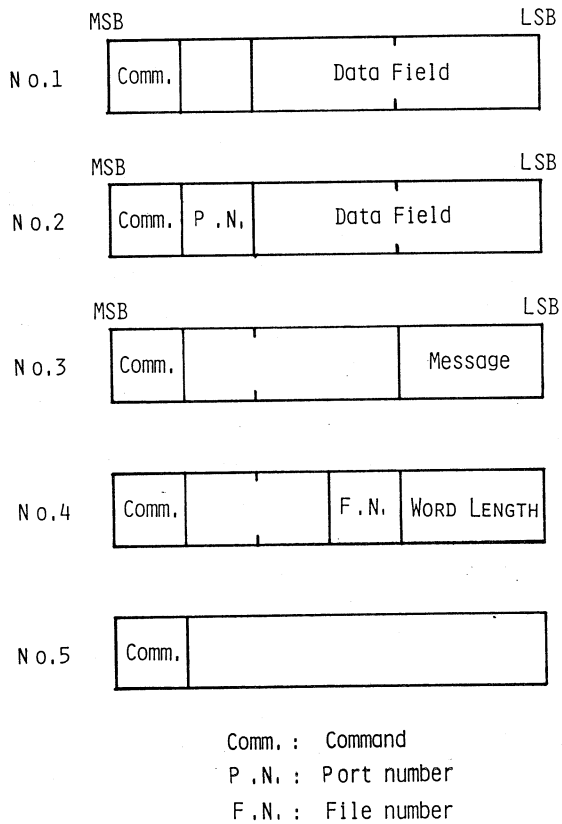


Fig. 3 Instruction formats for CIM/DIM system.

The first format is used for testing the fundamental functions of CIM and DIM. The second format is used for writing/reading data to/from one port or two ports of a DIM. Using the instruction format, the control computer can communicate with 256 input and output ports of each CIM/DIM system. The third format is used for starting a macro-program. A macro-program is specified by the least significant byte as a message. Therefore, 256 kinds of macro-programs can be defined at each CIM/DIM system. The fourth format is used for reading out the file-information taken by the timer task, where the file number is specified by the lower four bits of the middle byte. The last format is used for initializing CIM and DIM, reading out a message which appears at the end of the execution of a macro-program, at an apparatus failure or at a failure of communication devices between CIM and DIM.

The above each instruction formats are sent to the CIM/DIM system through CAMAC write data-lines (W_1-W_{24}). Subaddress (A_1-A_8) is used for specifying a DIM. The function command of the CAMAC's, F(n), are defined as follows.

F(0): reads out CIM/DIM system-information in a single mode.
F(1): reads out CIM/DIM system information in a repeat mode.
F(9): restarts CIM.
F(10): clears LAM-flag of CIM.
F(16): writes control computer-information into CIM/DIM system.
F(20): interrupts CIM/DIM system.

COMMAND FUNCTION

Table 1 summarizes command codes specified in an instruction, the name of each command, and the number of the applied instruction format. The function of each command is as follows:

(1) CIM Initialization
A CIM executes its initialization program.
(2) DIM Initialization
A DIM executes its initialization program.
(3) CIM Test
CIM fetches information from the CAMAC write data-lines (W_1-W_{24}) and sends it back to the CAMAC read data-lines (R_1-R_{24}).
(4) DIM Test
CIM/DIM system fetches information of the CAMAC write data-lines (W_1-W_{16}) and send it back to the CAMAC read data-lines (R_1-R_{16}).
(5) One Byte Data Write in One Port
CIM/DIM system fetches information of the CAMAC write data-lines (W_1-W_8) and writes it in one port specified by port number (P.N.).
(6) Two Bytes Data Write in Two Ports
CIM/DIM system fetches information of the CAMAC write data-lines (W_1-W_{16}) and writes it in two ports (P.N. and P.N. + 1) specified by a port number (P.N.).
(7) Message to CIM/DIM
The command is used for starting a macro-program. The content of a message is indicated by the least significant byte of the instruction format. Representative macro-programs are:

(A) Timer Task Start.
(B) Block Data Transfer from CIM Memory to DIM Memory Start.
(C) Block Data Transfer from DIM Memory to CIM Memory Start.
(D) Probe Measurement Start.
(E) Magnet Current Control Start.
(F) RF System Control Start.

(8) Block Information Write in CIM Memory
CIM/DIM system stores the block information of the control computer in CIM memory.
(9) One Byte Information Fetch from One Port
CIM/DIM system fetches the information of one port with a port number (P.N.) and sends it to the CAMAC read data-lines (R_1-R_8).
(10) Two Bytes Data Fetch from Two Ports
CIM/DIM system fetches the information of two ports (P.N. and P.N. + 1) by specifying a port number (P.N.) and sends it to the CAMAC read data-lines (R_1-R_{16}).

Command		
Code	Name	Format No.
0000	CIM Initialization	5
0001	DIM Initialization	5
0010	CIM Test	1
0011	DIM Test	1
0100	One Byte Data Write in One Port	2
0101	Two Bytes Data Write in Two Ports	2
0110	Message to CIM/DIM	3
0111	Block Information Write in CIM Memory	2
1000	One Byte Information Fetch from One Port	2
1001	Two Bytes Information Fetch from Two Ports	2
1010	Analogue Data Fetch	3
1011	One Byte Information Fetch from DIM Memory	4
1100	Two Bytes Information Fetch from DIM Memory	4
1101	Block Information Fetch from CIM Memory	2
1110	Message from CIM/DIM	5

Table 1 Commands for CIM/DIM system.

(11) Analogue Data Fetch
DIM samples all the analogue input data at a time interval, digitizes them with a 12-bits A/D converter sequentially and stores the digitized information into their respective Memory. CIM/DIM system fetches the information of memory with a port number (P.N.) and sends it to the CAMAC read data-lines (R_1-R_{12}).

(12) One Byte Information Fetch from DIM Memory
The command is used for fetching the information obtained by the timer task. CIM/DIM system fetches the one byte information of a DIM's memory area with a file number (F.N.) and sends it to the CAMAC read data-lines (R_1-R_8) sequentially, where the word length of the sent information is specified by the least significant byte.

(13) Two Bytes Information Fetch from DIM Memory
The command is used for fetching the information obtained by the timer task. CIM/DIM system fetches the two bytes information of a DIM's memory area with a file number (F.N.) and sends it to the CAMAC read data-lines (R_1-R_{16}) sequentially, where the word length of the sent information is specified by the least significant byte.

(14) Block Information Fetch from CIM Memory
The command is mainly used for fetching the block information obtained by a data logging. CIM sends out the block information in a predetermined memory area of CIM to the CAMAC read data-lines (R_1-R_8) in the unit of 4 K bytes sequentially.

(15) Message from CIM/DIM
The message is used for fetching the cause for appearance of a LAM-flag and sending to the CAMAC read data-lines (R_1-R_8).

REFERENCES

1 K.Shimizu, J.Fujita, T.Wada and I.Yokoyama: RIKEN Accelerator Progress Report, vol.18 (1983),p141.
2 K.Shimizu, I.Yokoyama, T.Wada, J.Fujita and Y.Yano: Symp. on Future Project of RCNP, (1981), p67.
3 K.Shimizu, T.Wada, J.Fujita and I.Yokoyama: Tenth Int. Conf. on Cyclotrons and Their Applications, (1984)