UDC 681. 322. 06

SOFTWARE SYSTEM IN FACOM SERIES FOR PROCESS COMPUTER CONTROL

Tsuneyoshi Ōboshi

Hiroshi Shirae

Control Technique Center

I. INTRODUCTION

The process control use software systems currently available and their corresponding central processing units are listed below:

1)	FACOM 270–30	MONITOR III_2 and MONITOR III_3
2)	FACOM 270–25	ROSP
3)	FACOM 270–20	MONITOR III_2 and MONITOR III_3
4)	FACOM R	PCPS
5)	FACOM 270–10	MONITOR I

II. MONITOR III_3 SYSTEM

1. Minimum Configuration

The MONITOR III_3 is suitable for use with the FACOM 270–20 and FACOM 270–30. The minimum configuration required to operate the MONITOR III_3 is given below.

1)	Central processing unit (FACOM 270–20/30)	1
2)	Core memory	16 kwords
3)	Magnetic drum memory	128 kwords
4)	Console typewriter	1
5)	Paper tape reader	1
6)	Paper tape puncher	1
7)	Real time controller	1
8)	Process control devices	1 set

Additional I/O devices, such as line printer, magnetic tape unit, external magnetic drum, XY plotter, cathode-ray tube character display, etc. may also be included. In this case, the capacity of core memory must be increased.

2. System construction

The construction of the MONITOR III_3 system is shown in *Fig. 1.*

1) The program control program performs interruption analysis to determine the causes of external interruptions, interruptions from the processor itself, program interruptions, and other interruptions. Priority control is performed with a maximum of 12 priority levels. The programs are divided into programs resident in core memory and programs which permit multiple use of core memory. The latter performs joining the system subroutine during execution and roll-out and roll-in service, segmentation service, and other services. In addition, it includes the dynamic load function of a system subroutine (primarily IOCS) for more effective use of main memory about the job processing program. Also, one area of core memory is provided for multi-use, but there is no multiprogramming function.

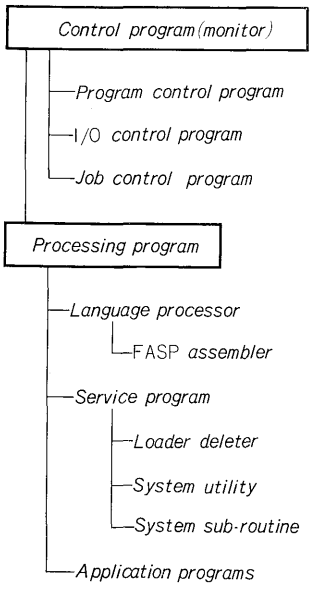

Fig. 1 Constsuction of MONITOR III_2, III_3 system

2) I/O control program

The program which processes I/O interruptions and the part related to the I/O control subroutine comprise a single unit and form an IOCS.

3) Job control program

There are two modes, batch operation in which a group of control records is processed continuously and instant operation in which individual instructions are punched in from the console typewriter. This program includes job interruption, restarting, forced end, and time monitoring function.

4) Error control program

Error consist of central processor errors, I/O device errors, software errors. Central processor and software errors are processed by the control program,

while I/O device errors are processed by the processing program through IOCS. The user can combine the independent error control program at system generation.

5) FASP assembler

The assemble uses subroutine pseudo instructions and macro instructions and instruction for subroutine, in addition to execution instructions corresponding to machine instruction are permitted the selection of the combining method of subroutine and segmented program operation are made easy.

6) FORTRAN compiler

The FORTRAN compiler includes JIS FORTRAN level 7000 and a segmentation functions, debugging functions, and fundamental arithmetic functions.

7) ALGOL compiler

The ALGOL compiler includes JIS ALGOL language 5000 and I/O level 50 and contains an abundance of fundamental functions.

8) System generator

The MONITOR III_3 system generator permits selection of any module of the control program as well as the combination of the user's program with the control program. In addition, a software system can be automatically constructed by inputing hardware and software control data.

9) Loader-deleter

This program loads and deletes the program and subroutine. Management of various program and subroutine control tables and management of memory accompanying loading and deletion are performed.

10) Utility grograms

These programs are essential in computer operation and are program preparation and debugging aids. A large number of system maintenance aids are available. The user can thus construct an ideal software system by selecting the best utilty for his system.

11) System subroutines

System subroutines are comprised of program management routines and I/O control routines. Each routine is constructed to permit selection of various optional functions to handle various processes.

12) Debugging program

This program is comprised of an on line tracer for on line debugging, dummy IOCS for various process control devices, data and address conversion routines, etc.

13) Applications library

A package for analog inputs which performs check-out, scaling conversion, updating, smoothing, and integration, a package for logging which simultaneously prints the data of any arbitrary address in the core or drum on a number of typewriters in any desired format, a gas chromatograph package and other process control inherent libraries, and LP, PERT/MANPOWER, and matrix operating packages are avilable.

III. MONITOR III_2 SYSTEM

The MONITOR III_2 system is identical to the MONITOR III_3 system except for the following points :

1) Minimum configuration

Core memory in the 8 kw～32 kw range (MONITOR III_3 core memory is in the 16 kw ～ 64 kw range). However, in the case of 8 kw, on line program preparation and testing is impossible.

2) System generation of the control program is impossible by manual operation.

3) No subroutine dynamic loading function.

4) I/O device logic No. specification is impossible.

IV. ROSP

1. Minimum Configuration

ROSP (Real time Operating System for Process Control) is suitable for the FACOM 270–25. Minimum configuration for ROSP operation is given below.

1)	Central processing unit (FACOM 270–25)	1
2)	Main memory (core)	24 kwords
3)	Magnetic drum memory	256 kwords
4)	Console typewriter	1
5)	Paper tape reader	1
6)	Paper tape punch	1
7)	Real time controller	1
8)	Process control devices	1 set

As described in the hardwere item, all types of I/O devices can be connected; however, memory capacity must be increased as required.

2. System Construction

ROSP system construction is shown in *Fig. 2.*

1) TASK control program

ROSP performs multiprocessing of programs. In this case, the processing unit is called a task. Up to a maximum of 63 tasks can be handled. Each task is assigned one of 8 priority levels 0～7. In other words, eight level priority control and parallel control of up to 63 tasks are possible. The task control program selects the highest priority task which can be executed and executes it. Programs which cannot be executed are waited. In addition, various task services are also performed (*Table 1*). Special care is required with programs which demand high-speed processing, since the program start time may be shift by the state of operation of other tasks. For this reason, only one priority level is provided above the normal task. This level can interrupt processing even during running of the program control program.

2) I/O control program

Since the speed of the I/O devices is generally slower than that of the central processor, a control

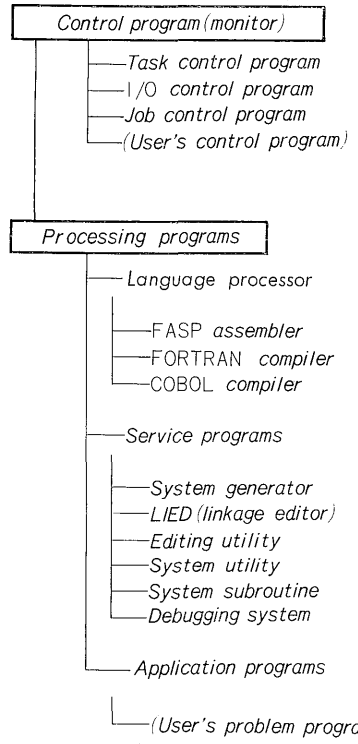

Fig. 2 Construction of ROSP

Table 1 ROSP service routine for program control

Start/stop of task
ENTER
RETURN
Synchronization between task
WAIT
SWAIT (Single Wait)
POST
ENQ (Enqueue)
DEQ (Dequeue On)
QOFF (Queue OFF)
MON (Mask On)
MOFF (Mask OFF)
Program control
LOADX (Load and Transfer Control)
Interrupt control
SPIE (Specify Program Interruption Exit)
RESET
STIMER (Set Timer)
TTIMER (Test Timer)
Special control of task
ABEND (Abnormal End)
CENTER (Cancel Enter)
SENTER (Stop Enter)
PENTER (Re-Enter)
SLEEP
WAKE
Stop control of system
STOP
Special service
SVC
RTNO (Read Task NO)
EXOCAL
RMASK (Read Mask)
WMASK (Write Mask)
ECBR (ECB Read)
ECBI (ECB Initialize)
Synchronization for parallel running
PENT (Parallel Enter)
PWAIT (Parallel Wait)
PRET (Parallel Return)

Table 2 ROSP service routine for I/O control

Assignment and release of I/O device connected to channel
UNITA (Unit Assign)
UNITR (Unit Release)
Special control of I/O device connected to channel
UNITC (Unit Control)
Access to I/O device connected to channel
READ
WRITE
CONT (Control)
RDW (Read Drum Word)
WDW (Write Drum Word)
Access to I/O device conneted to direct line
DIA (Digital Input Type A)
DIC (Digital Input Type C)
DOA (Digital Output Type A)
DOB (Digital Output Type B)
DOC 1 (Digital Output Type C 1)
DOC 2 (Digital Output Type C 2)
CLK (Real Time Clock)
PCI (Pulse Counter Input)
AI (Analogue Output)
POS (Pulse Width or Train Output Set)
POR (Pulse Width or Train Output Read)
TIMS (Timer Set)
TIMR (Timer Read)
RTCTP (RTC-Typewriter or RTC-Paper Tape Puncher)
Console control
WTO (Write to Operator)
WTOR (Write to Operator with Reply)
System I/O
INPUT
LIST

program is required for efficient system operation. In addition, when progams for different jobs require the use of the same I/O devices, assignment of the I/O devices must be controlled. These functions are performed by the I/O control program. When an error occurs, this program diagnoses the trouble, notifies the task, and performs error processing (*Table 2*).

3) Job control program

The job control program controls the system I/O devices and operates the system in accordance with the information from the system input (*Table 3*).

The job control program constantly monitors execution of the job and permits on-line debugging of batch jobs.

4) FASP assembler

This is a program language for cases when extremely precise programming technique are required and can be concisely expressed using easy to remember words. It has an especially large number of macro instructions, and program division and linking functions.

5) FORTRAN compiler

The FORTRAN compiler is a newly developed language for preparation of real time use programs. The compiler is based on JIS FORTRAN level 3000 supplemented by level 7000 functions and has the following additional features:

(1) Double integer

(2) Format conversion without I/O

Table 3 **ROSP job control record**

Control sentence	Descrription	Console input	System input
¥ * □ Comment	Comment	○	○
¥ SYSIN □ ddnn	Change of system input	○	○
¥ JOB	To start the job		○
¥ Segment name □ Parameter	To start the program	○(1)	○
¥ □ Parameter	Continuation of the above		○
¥ END	End of system input	○	○
¥ KILL	To kill the batch job	○	
¥ PAUSE	To pause in the batch job	○	
¥ PROC	To restart the batch job	○	
¥ NOCVT	To stop converting code		○(2)
¥ ENDNC	To restart converting code		○(2)
¥ DATE □ yy.mm.dd	To set the date	○	
¥ TIME □ hh.mm.ss	To set the time	○	

Note (1) Only in real job
(2) Only in paper tape device

(3) Access of external symbol
(4) Generation of core resident program
(5) Overlay construction of program
(6) Definition and direct access of real data file
(7) Debugging statement
(8) Abnormal return

6) System generator

The system generator comprises a control program generator and a processing program generator. The control program generator generates a system ideally suited to the user's needs by selecting the specified parts from the body of the control program and modifying them. The control program is designed to permit a large number of modules in core resident or drum resident, and can be effectively used both by user's desiring preparation of a large core resident high-speed system, and by user's desiring to use the ROSP functions at low speed. The processing program generator is available for a system program such as program language and utility, and user prepared programs.

7) LIED (linkage editor) is a program which links and generates relocatable binary program elements which are translated by the assembler and compiler and prepares a relocatable binary program of segment units.

8) The editting utility performs registration, deletion, renaming, listing, and updating of the source library, object library and system library. It also performs registration and dumping of the various system tables.

9) The system utility includes a magnetic tape dump, disk pack dump, and other similar programs.

10) The system subroutines are a group of routines which permit common use of the process PID arithmetic routine, FCP, subroutines for various science, arithmetic etc. in the processing program and application program.

11) The debugging system is a debugging utility and includes a tracer and a snap shot dump RTC dummy subroutine, etc.

V. PCPS

1. Minimum Configuration

PCPS (Process Control Program System) is a software system aimmed at the FACOM R.

1) Central processing unit (FACOM R)
2) Core memory 8 words
3) Magnetic disk cartridge 64 kwords
4) Console typewriter 1
5) Paper tape reader 1
6) Paper tape puncher 1
7) Real time controller 1
8) Process control devices 1 set

PCPS can be used with a core memory of 4 kword or greater in a small scale system without using a magnetic disk cartridge, but this explanation is omitted here.

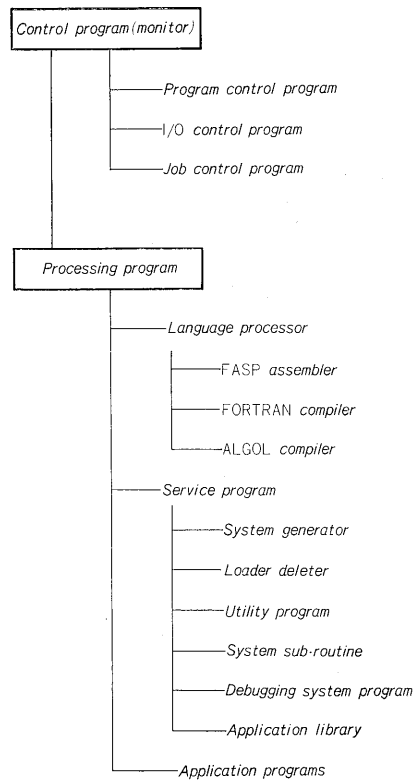

Fig. 3 **Construction of PCPS**

2. System Construction

The construction of the PCPS system is shown in *Fig. 3.*

1) Program control program

Priority processing and parallel control are performed in PCPS. There are a maximum of 15 priority levels. The scale of parallel processing depends upon the size of the register save area in core memory.

2) I/O control program

Since the speed of the I/O devices is generally slower than that of the central processor, the control program must be controlled for efficient system operation. In addition, when different programs require the same I/O devices, a control program must manage the I/O devices as a total system and control their order of use, etc. The I/O control program performs these controls. When trouble occurs, the program diagnoses the trouble, notifies the program, and performs error processing.

3) Job control grogram

The job control program operates the control of the system I/O devices and the operation guide of the system in accordance with the information from the system input, and is primarily used for program preparation and testing.

4) FASP assembler

The FASP assembler concisely represents machine language in easy to remember words and has simple pseudo instructions.

The FACOM R is an ultra-small computer and ample I/O connection for program preparation and testing is difficult. Moreover, its processing speed is slow. Therefore, we have prepared a FACOM R assembler and registration system as a MONITOR III_8 system or ROSP utility. The use of these permits preparation of an efficient program with abundant MONITOR III_3 and ROSP fnnction.

5) The program loader-deleter loads the core imaged program after translation by FASP and deletes a loaded program.

6) The system utility includes memory dump rewrite, system state dump rewrite, FASP source program modify, and other programs.

7) The system subroutines include various arithmetic subroutines, disk cartridge file control subroutine, process I/O file control subroutine, process control use subroutine, etc.

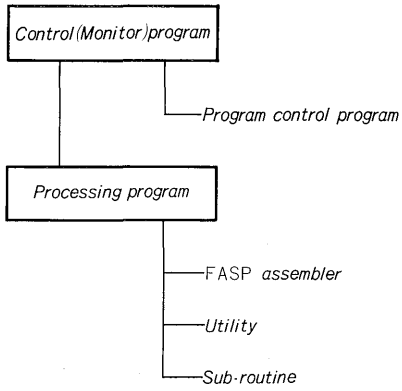

Fig. 4 Construction of MONITOR I system

VI. MONITOR I SYSTEM

MONITOR I is a softwere system aimmed at the FACOM 270–10 (refer to *Fig. 4*).

1. Minimum Configuration

1)	Central processing unit (FACOM 270–10)	1
2)	Core memory	2 kwords
3)	Typewriter	1
4)	Real time controller	1
5)	Process control devices	1 set

2. System construction

The construction of the MONITOR I system is shown in *Fig. 4.*

VII. CONCLUSION

A description of the process control use softwere system has been given above. Due to space limitations, a complete description was impossible. For details refer to the various explanation sheets and order documents.