The ACS-3000 Advanced Controller

Akira Nishikawa
Kouji Kawaguchi
Nobuo Taguchi

1. Introduction

The EIC-integrated system MICREX-IX, released in 1992, is the fourth generation of the distributed control system (DCS) developed by Fuji Electric. Many of these systems have been adopted as the core of automation systems in various industries. Now, the advanced information and control system MICREX-AX has been developed, a further extension of the original system.

The concept of EIC (electric, instrumentation and computer) integration is the realization of a closely knit combination of the product and its production process. For the DCS, this requires an integration of the product-oriented computer and the process-oriented controller.

The controller of the MICREX-series adopts a standardized architectural formula, achieving simplified EIC integration through a hierarchical arrangement of the function boards, package software and network, based on common technologies of the hardware and software.

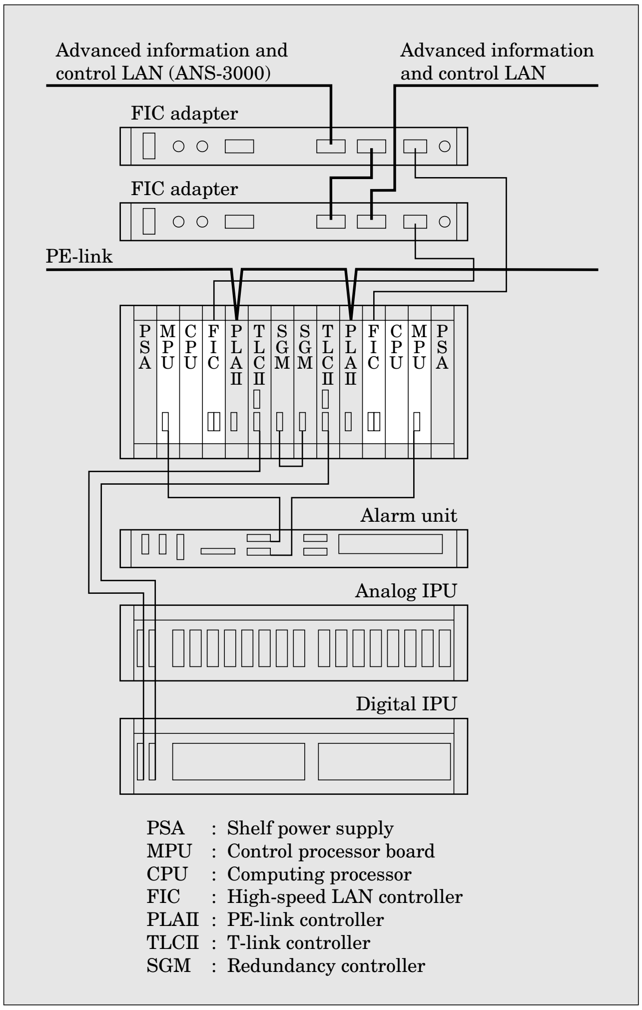
The newly developed advanced controller "ACS-3000" aims to realize an optimal system construction. This was achieved minimal engineering costs and by fusion of the computing function into the controller and mounting integrated data handling and support systems on its standardized architecture.

The features of the system include:

(1) a "multiprocessor" structure, which has a control processor together with a computing processor
(2) a high-speed computer connection via the LAN, based on the international standard FDDI (fiber distributed data interface)
(3) a "controller data handling system" based on the data name and physical value
(4) an engineering system that supports the above features

In this paper, the architecture and its accompanying basic technologies are described for the ACS-3000, which realizes effective engineering.

Fig.1 Basic hardware configuration of the ACS-3000



2. Hardware

Figure 1 shows the basic hardware configuration of the ACS-3000, and Fig. 2 shows the external view of its MPU shelf.

The ACS-3000 has the same architecture as the controller of the conventional MICREX series and can be connected commonly to the remote PIO (IPU, CIO, TK-capsule) via a T-link and directly to the PIO, P-link and PE-link.

2.1 Features

The ACS-3000, with its basic concept of flexibly constructing various systems with a combination of common components, is structured so that function unit boards are mounted on a 19-inch shelf. This shelf is then installed in a special cabinet. In addition, a stand-by redundancy system is achieved with a single shelf.

International interchangeability at the board level is achieved by utilizing a MULTIBUS II[*1], a 32-bit system bus (ANSI/IEEE1296).

2.2 Hardware architecture

Figure 3 shows the function block of the ACS-3000. Various types of independent boards for each function can be connected to construct an optional control system if needed. The multiprocessor configuration having a controlling processor board and a computing processor board enables the realization of multiprocessing or parallel processing of control and data processing.

2.2.1 Control processor board

The control processor board (MPU) executes the functional control language (FCL), an intermediate language between the application programs and the controller. We developed 32-bit FCL chip in which the hardware executes not only sequential and arithmetic instructions but also floating point arithmetic operational instructions. The chip is more than twice as fast as the conventional one.

Table 1 shows the control specifications of the ACS-

∗1 MULTIBUS II : A registered trademark of Intel Corp., USA

Fig.2 External view of MPU shelf

Table 1 Control specifications of the ACS-3000

Item		Specifications
Instruction	Language	Functional control language (FCL)
	Instruction processing speed	Sequence instruction : 0.08 to 0.52μs Fixed-point addition and subtraction : 0.12 to 0.56μs Fixed-point multiplication : 0.16 to 0.64μs Fixed-point division : 4.00 to 4.44μs Floating-point addition and subtraction : 0.16 to 0.64μs Floating-point multiplication : 0.16 to 0.64μs Floating-point division: 1.40 to 1.84μs
Memory capacity	Program memory	512k Byte (128k step)
	Data memory	1M Byte (512k word)
Number of I/O points	Digital	T-link I/O For all IPU configurations : 8,192 points For all CIO configurations : 4,096 points Directly connected I/O : 512 points Total number : 8,704 points
	Analog	T-link I/O For all IPU configurations : 2,048 points For all CIO configurations : 2,048 points Directly connected I/O : 128 points Total number : 2,176 points
	Timer	1,024 points
	Counter	128 points
	Keep relay	1,024 points
	Differential relay	1,024 points

Fig.3 The function block of the ACS-3000

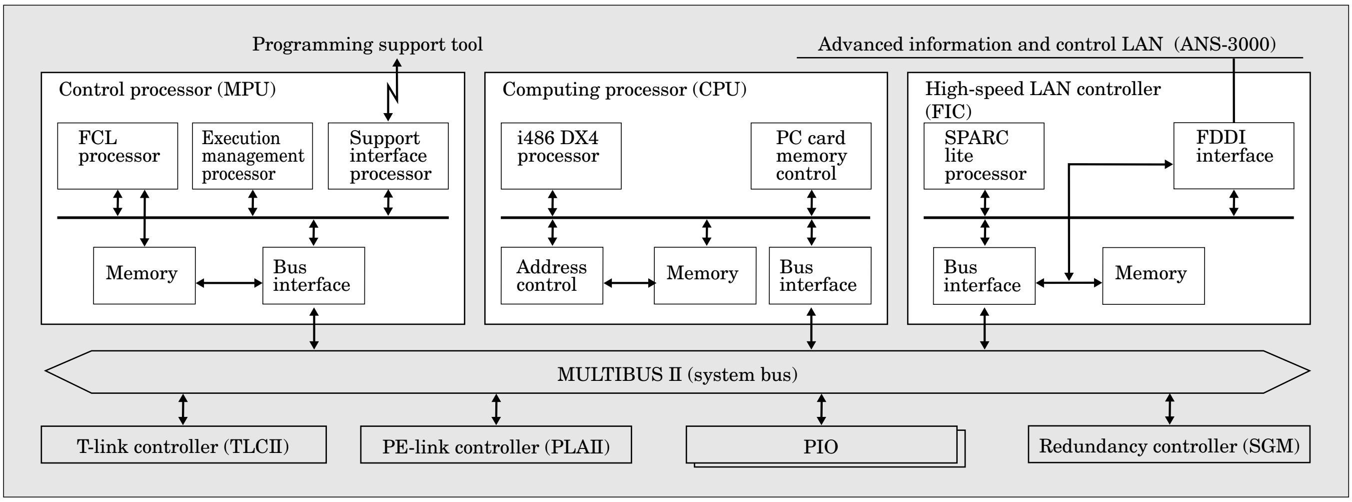
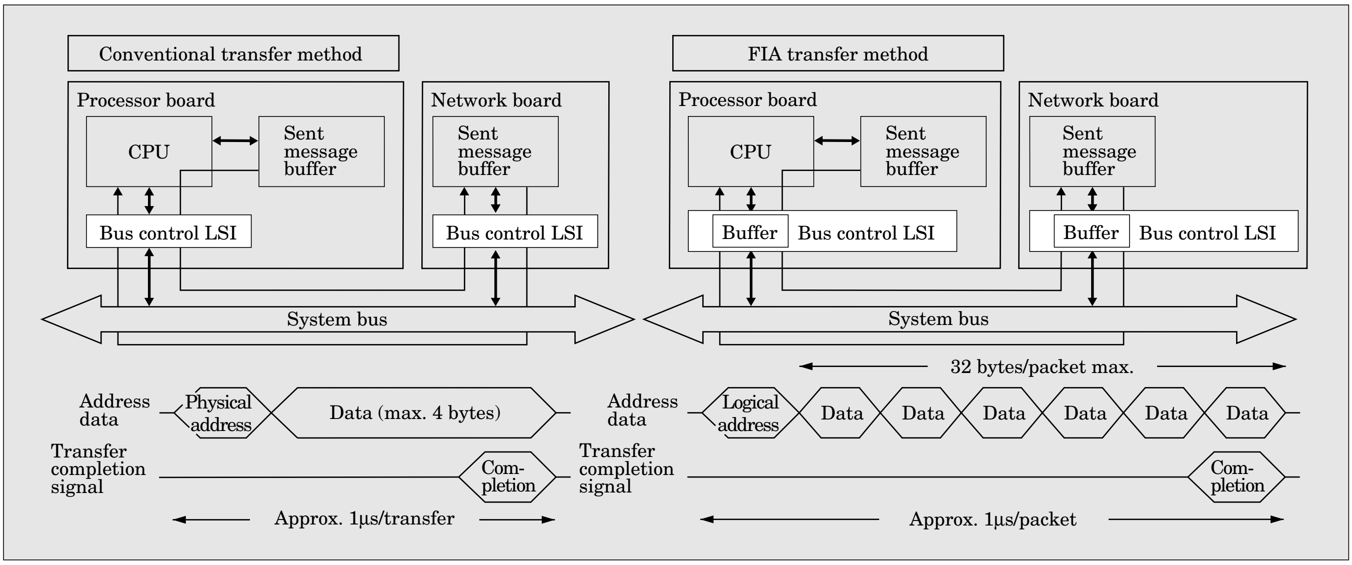


Fig.4 Data transfer method of the bus



3000.

2.2.2 Computing processor board

The computing processor board (CPU) uses the INTEL 486DX4, which is suitable for data processing. The micro kernel technique is adopted for its OS, enabling a real-time function by means of a "multi-thread" method.

In addition, two PC card memories can be mounted, which can be used for storing application programs as well as data names and physical value definitions.

2.2.3 High-speed LAN controller board for process control

The high-speed LAN controller board for process control (FIC) is a high-speed LAN controller for process control based on and extended from the FDDI, with a speed of 100 M-bits/second. By using the FDDI, high-speed communication to the advanced operator system AOS-3000 using a workstation and open connection to a computer system has become possible.

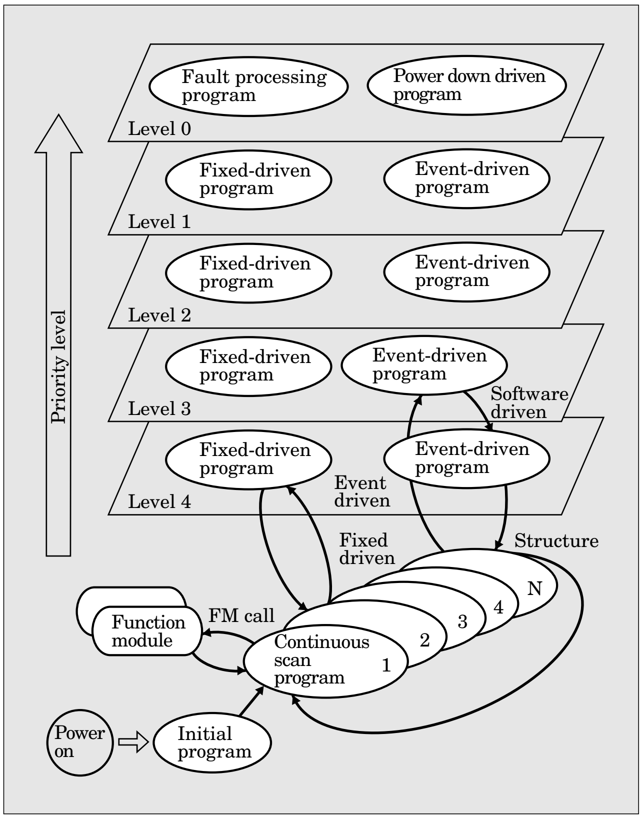
For further information on the high-speed LAN controller, please refer to the article "The ANS-3000 Advanced Information and Control LAN" in this same special issue.

2.3 Flexible system bus (FIA)

The FIA protocol greatly increases system bus transfer capability of the control processor board, the computing processor board and the network control board. The bus control LSI which executes this (MBC chip) has also been developed.

The data transfer method of the FIA (flexible interconnection architecture) is shown in Fig. 4. Within the time of transferring 4 bytes (32 bits) of data by the conventional physical transfer, 32 bytes of data are transferred by the FIA method. The transfer capability of the system bus has increased 8 times, ensuring sufficient capacity of bus transfer for the large data transfer volume between the multiprocessors and from network control boards.

Fig.5 Outline of program execution management



3. Application Interfaces

3.1 Programming language and program execution management

(1) Programming language

The programming language FCL has been used for the conventional MICREX series and can also be used for conventional software properties. FCL can develop various programming expressions including ladder diagrams, FB charts and SFC according to the control purpose. It is optimal as the intermediate language between the controller and the programming support system.

(2) Execution management of application programs

Program management, which controls execution of application programs in real time according to time sequence or to situation, is an essential factor in determining control function. An outline is shown in Fig. 5.

Program execution management controls four kinds of execution-continuous scan, fixed cycle driven, event driven and software driven-according to their priority levels.

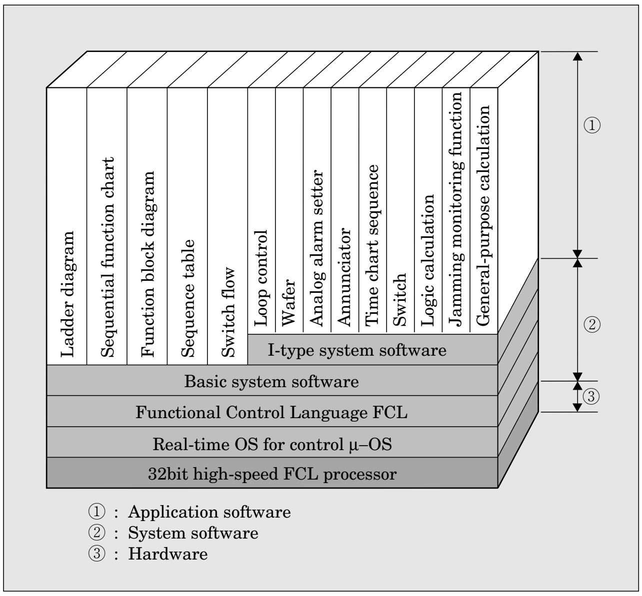
Also, the execution ratio of electric control (E), instrumentation control (I) and applications can each be set optionally, so that various control systems can be flexibly constructed. As programs can be executed as function modules, repeated use is simplified.

4. System Interface

4.1. System software

As shown in Fig. 6, the system software of the ACS-3000 is structured hierarchically, with instrumentation system software (I-type) packages mounted on the basic system software. Therefore, the ACS with only its basic system software operates as an electrical control (E-type) programmable controller (PLC). By adding the I-type software package, it then becomes an electrical and instrumentation (EI-type) controller. The control functions used by the I-type include loop control (internal instruments), analog alarm setter, annunciator, time chart sequence, switches, logic calculation and jamming monitoring function.

Fig.6 Software classes of the ACS-3000



4.2 Data management

Regarding the process data to be controlled by the controller, the purposes of data management are:

(1) to have the definition for control (the physical value definition), so that the application programs using the physical value definition can operate process data in unity

(2) to have the definition for process management by the operator (the attribute definition), so that the operator station using the attribute definition and physical value definition can indicate, operate and store process data in unity

In this system, the adding of the data management function to the ACS has enabled unified management, unified operation and access to the process data definitions within the system that includes the operator station AOS-3000.

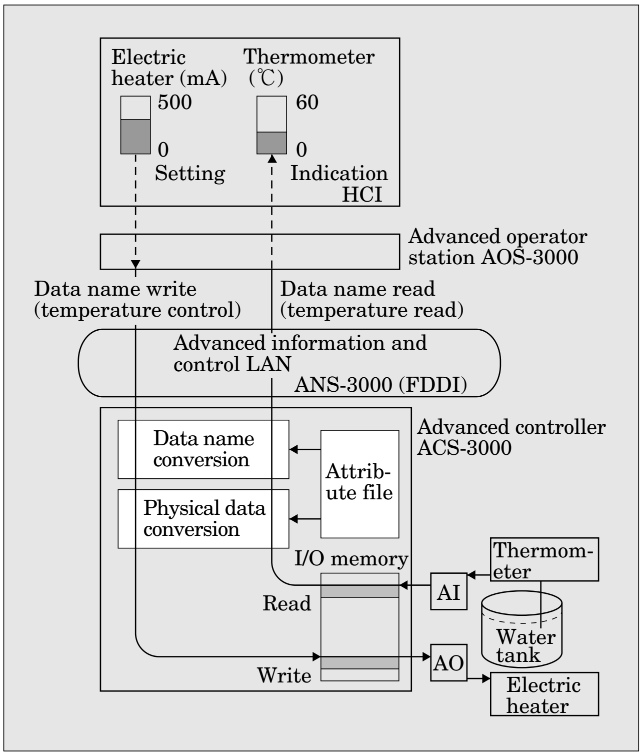
This has achieved a revolutionary improvement in system engineering efficiency. Moreover, as they are unified in the physical data definition also on the network, open interface with the computers has become possible.

4.2.1 Unified data management

Data management is defined by the engineering support system (AES-3000 or FPROCES-C). It is function within the ACS-3000 that manages in unity unique data names (tag number and label), attribute value definitions (comments and messages) and physical value definitions (unit, scale and unit quantity).

Under unified data management, the AOS-3000 accesses process data after first up-loading definitions

Fig.7 Data name access method



of the controller. Even for often executed changes such as scale, simultaneous changes throughout system are enabled by changing only the attributes of the ACS form the support system.

4.2.2 Physical value operation

In the ACS-3000, the physical value definitions can be set optionally for input/output data, internal data and instruments. As a result, the application programs can execute their process data operations without regard to the data type. Consequently, function blocks with physical value, specified by the fieldbus, can be easily programmed.

Furthermore, since both conventional %-data and floating-point data are treated as internal data formats, it is possible to utilize existing software properties as they are.

4.2.3 Data name access

The data name access function enables the AOS-3000 access by indicating the unique data name within the system defined by the process data.

With data name access, the AOS can broadcast and send access requests to every ACS-3000. Each ACS, after checking to see if the corresponding data name exists within itself, can send back its result, if any, to the requesting source. An example of data name access is shown in Fig. 7.

In the data name access method, equipment of a higher class such as the AOS and computer systems can be accessed by merely using its data name. Because the ACS executes access-operation to find the data name from its own data name definition, equipment of a higher class does not need to know , to which ACS the data belongs. Furthermore, the data name is unique within a system. When the data arrangement is changed within the system or between systems, it is only necessary to change the data name definition of the ACS. No change of higher class application programs is required.

4.3 HCI interface function

The ACS-3000 has been installed with functions conventionally executed by an exclusive database stations. These functions include storage of historical data such as state change and trend and HCI-interface function such as a system clock. The ACS and AOS-

Fig.8 State transition

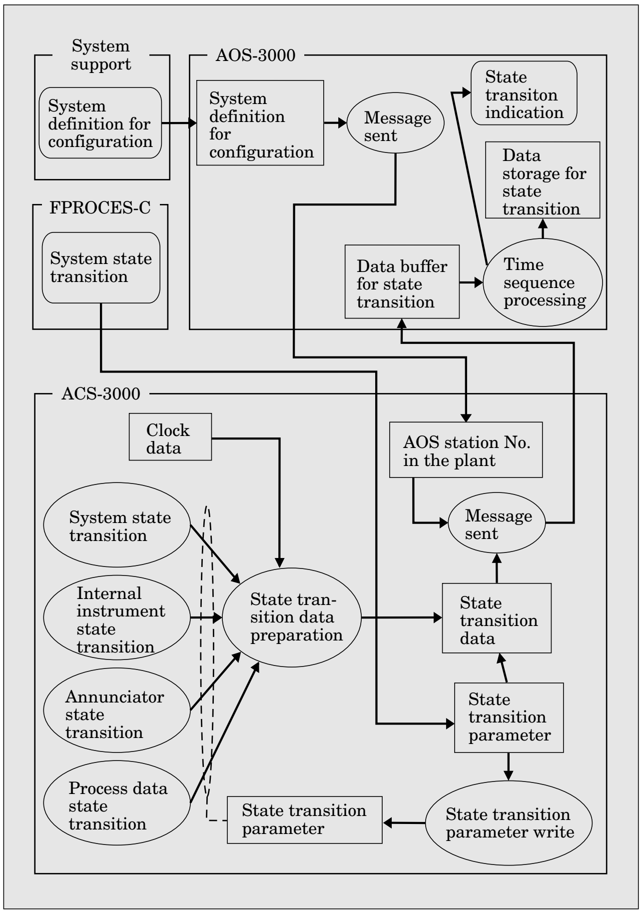


Fig.9 Usual trend / batch trend function

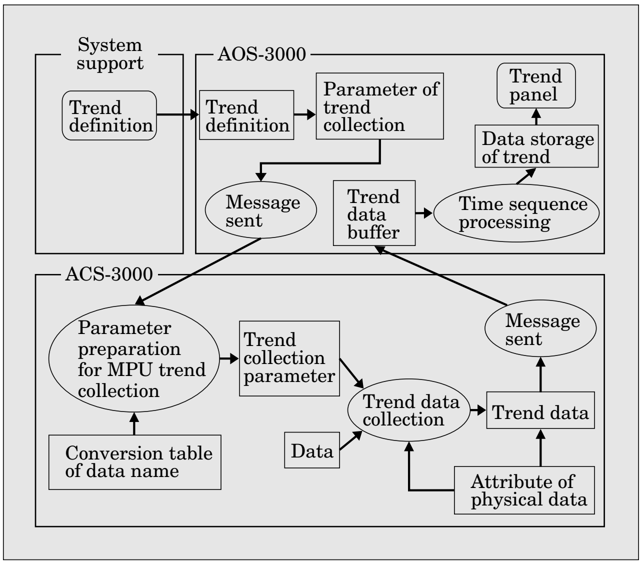
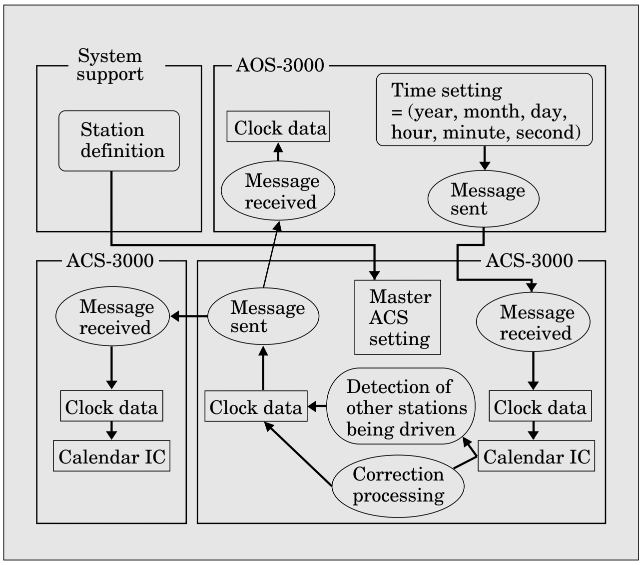


Fig.10 System clock function



3000 are solely capable of configuring simple yet complete system.

4.3.1 State change function

The ACS-3000 is able to detect state changes of process data and alarm signals in the internal instrument and can send this information, attaching the system clock time, to the AOS-3000. As a result, it can store and indicate the history of state transitions in the entire system, along with their correct times occurrence. (Fig. 8)

The state change function of the ACS is divided into four types and attached with its occurrence time in milliseconds:

(1) System state transition

Its function is to report the system transitions in the ACS and contains the following details:

(a) normality/abnormality of such boards as the TLC II, PLA II and SGM
(b) normality/abnormality of the PIO
(c) normality/abnormality of such instruments as the FCX transmitter and FFI equipment

(2) Internal instrument state change

This reports occurrence/return of alarm signals of the internal instruments (PID controller, alarm indicator, etc.)

(3) Annunciator state change

This reports occurrence/return of the annunciator output.

(4) Process data state change

This reports the positive/negative edge of bits.

4.3.2 Trend function

As the ACS-3000 samples and stores process data and PIO data and sends them to the AOS-3000, it can execute correct sampling during the data collection cycle or at the collection's start/stop signal. (Fig. 9)

There are two kinds of trend functions, normal-trend and batch trend. The normal trend collects data according to the set cycle, and the batch trend collects data at each collection's start/stop signal.

4.3.3 System clock function

In this system, a master system clock is built into a specified ACS-3000. By coordinating the synchronized time correction of the ACS and AOS-3000, the system time lag for each system is kept to a minimum. (Fig. 10)

(1) Time setting function

The AOS-3000 sets time data (year, month, day, hour, minute and second) into the ACS-3000, which contains master system clock. The master ACS with the set time sends the time data to the other ACS and AOS and sets system clocks for all stations. The clocks of the stations which have been driven during system operation are also set.

(2) Clock correction function

Once the clocks have been set, the master ACS transmits the clock's correction data in a fixed cycle in order to correct the system clock of each station.

Each station corrects its own system clock using the transmitted clock correction data. The clock correction function keeps the time lag between station to tenths of a millisecond.

5. Conclusion

The aims, architecture and basic technologies of the ACS-3000 advanced controller are described in this paper. The architecture of the ACS-3000 is extensible due to the fact that it takes technological development into consideration. We plan to propose systems which support connection of the ACS to various networks and equipment, including fieldbuses.

We propose to develop software packages for various application areas including advanced control for rational development of applications and to develop engineering tools for integrated support.

Moreover, we will continue our efforts, kindly supported by our customers, to further improve our systems.