MIMI '85 SANT FELIU
SPAIN

ISBN 7488-121-

MINI AND MICROCOMPUTERS AND THEIR APPLICATIONS

Editor: E. Luque

A PUBLICATION OF
The International Society for Mini and Microcomputers (ISMM)

UNIVERSIDAD AUTONOMA DE BARCELONA

Area: Networks

Area: Education

Area: Education

INDUSTRIAL PROGRAMMABLE ADAPTIVE CONTROLLER: HARDWARE AND SOFTWARE STRUCTURE

S. Dormido[*], J. M. de la Cruz[**], H. Ruipérez[**], F. Morilla[*], J. M. Guillén[***]

[*] Dpt. de Informática y Automática, Facultad de Ciencias. UNED. Madrid.
[**] Dpt. de Informática y Automática, Facultad de Física. Univ. Complutense. Madrid.
[***] Instituto de Electrónica de Comunicaciones. C.S.I.C. Madrid.

Abstract

This paper presents an industrial programmable controller which includes adaptive control. From the hardware point of view the controller has been designed to be compatible with most current commercial regulators. Nevertheless it also allows the implementation of new control methods. The controller has the possibility of being used as a single unit controller within a distributed control structure.

The software architecture of the digital controller is also presented. There exists two differents parts. The first one reads-writes and logs the input-output signals of the controller, the second one calculates or modifies the control value. The software has been designed to allow all inputs-outputs and control functions be externally configured by means of a programming panel included in the controller.

INTRODUCTION

As it is very well known, the introduction of the microprocessors in control instruments is changing the control process market. Many industrial applications have made a transition from analog to digital implementation. Microprocessors provide the computing power for this evolution. Among the main contributions they provide to process control are:

- Making more versatile controllers in the sense that, by means of programming, the same regulator can be used in many different processes.

- Increasing the complexity of the control algorithm, since they permit the application of almost any control algorithm to real processes.

- Increasing the reliability of large systems through distributed control.

- Shrinking the size and cost of many computer control systems, so that smaller processes can use computer control.

Most general purpose control systems designed for use on large number of processes systems use different kinds of PID algorithms (1). That is mainly due to the fact that, as the experience has shown, for most industrial processes good control performance can be achieved by properly tuned PID controllers However, the computing power of microprocessors is helping to reduce the existing gap between theory and practice in Automatic Control. Thus, there have also been developed general purpose control systems which use more complex controllers (2), (3). Special effort has been devoted to getting adaptive controllers, sometimes referred to as self-tuning controllers or self-optimizing controllers.

Although the feedback performance of PID controllers is well known, in the world of practical loads, controller tuning can be difficult and time-consuming. Therefore the broad use of PID controllers in industrial control processes makes PID parameters auto-tuning a fundamental task. Recently some general purpose self-tuning PID controllers have been developed (4), (5).

The controller we present in this paper is an industrial programmable controller based on the 6809 microprocessor. The most important characteristic of the controller is the existence of the adaptive mode, in addition of the traditional manual, automatic, and computer In the adaptive mode, as the process control evolves, the PID parameters can be modified to the end that the process control be improved. Moreover, the programming facility allows the controller be configured according to the process that is being controlled. The controller has been designed to be compatible with most PID controllers existing nowadays; in that way, its introduction in a control loop may be made by direct substitution of one controller by the other.

1. HARDWARE SPECIFICATIONS

The controller is based on a 8-bits microprocessor and may be divided into four basic units (Fig. 1).

U1-Input-Output Unit
U2-Central Process Unit
U3-Keyboard and Displaying Unit
U4-Signal Conditioning Unit

1.1 Input-Output Unit

This unit encloses the analog and logic inputs and outputs of the system with the following distribution: 4 multiplexed differential analog inputs; 3 analog outputs; 8 incremental inputs; 8 logic inputs and 4 logic outputs.

Analog inputs: The four differential analog inputs are multiplexed and joined to an Instrumental Amplifier whose output feeds a 12-bits A/D converter so that the system is able to accept signals of very low level.

Analog outputs: The three analog outputs are independent of each other. They allow monitoring two chosen process variables in addition to providing the control signal.

Although the system is thought to be monovariable, the four analog inputs and three analog outputs may be arbitrarily configured giving a great flexibility for process control.

Incremental inputs: There are logic signals associated to the analog inputs. Any pair of them may be assigned to an analog input. They are activated at low, pointing out either an increment or decrement of their associated variable. The increment or decrement in the associated variable is proportional to the time interval that the input signal has been activated.

Logic inputs: These logic variables may be either fixed to a given value by the user or changed by the process dynamic. They allow that events external to the process control change the way the system works.

Logic outputs: These variables are used by the system to show fixed aspects of the process control.

1.2 Central Process Unit

This unit encloses an 8-bit microprocessor, the RAM and ROM memory, the timing circuits and a RS-232 interface. There are two sorts of information stored in ROM:

a) Data base: Parameters which configurate the controller in its cold start;

b) Control Programs.

There are two different areas of RAM: a) The variables area; b) The data base. The latter is formed by those parameters of the ROM data base that may be modified by the user to definitely configure the system.

The RS-232 interface included in this unit gives the possibility of using the controller in distributed control. It also allows that a computer be able to exercise the control action via the computer mode of operation.

1.3 Keyboard and Displaying Unit

The system has an user panel and a programming panel. The main functions of the user panel are: a) to display the set point, the controlled variable and the control action; b) to modify the mode of operation (manual, automatic,computer adaptive); c) to modify the values of the set point and the control action. This last possibility only in the manual operation mode.

The programming panel gives the possibility of configuring the system with a desired structure.

1.4 Signal Conditioning Unit

The function of this unit is to isolate the system from external perturbations and to adapt the analog signals to the voltage ranges of the converters.

2. FUNCTIONAL DESCRIPTION: BASIC PROCESSES

The basic objective of including a regulator in a control loop is that of generating the control signal. In order to achieve this goal it is necessary, first of all, to fit the regulator for the process which is going to control, that is to say, all its inputs and outputs, the control law and internal control logic must be defined. We have call all these actions "configuration" of the regulator and it can be made through a specific keyboard built up in the regulator.

Once the regulator has been given a definite structure, the next step is to provide initial values to all its parameters. The regulator picks up these values from specific memory registers and they can be modified afterwards by the human operator if necessary. This is what we have called "start-up" procedure.

At this point the regulator is in condition to begin the control of the process. It reads the input signals, makes some characterizations in order to achieve linear behaviour of the internal variables, makes some tests to determine if there is a situation that needs an alarm to be triggered and control action interrupted, calls the routines that calculate a new control and sends this control to the output. We have called this long process "regulation" and it can be split into two small ones: a process of "reading-logging-writing" and a process of "control calculation".

3. DYNAMIC DESCRIPTION

From a dynamic point of view, all the above processes can be gathered in two kinds of processes:

a) Non cyclic: Configuration and start-up procedure.

b) Cyclic : Reading/logging/writing and control calculation.

Non cyclic processes: The regulator goes into these processes only under very particular circumstances, say, when first installing, when the parameters need to be changed or after a power failure.

Cyclic processes: Once the regulator has been configured and it has completed the start-up procedure, it goes into the cyclic processes above mentioned. Since there are two cyclic processes it is necessary to share the time between them.

The time constants of the industrial processes for which the regulator has been designed are quite large. It implies that the sample intervals will be at least of several seconds. In order that the monitoring of the input signals be efficacious, the regulator must go into the reading/logging/writing (R/L/W) process more frequently than into the control calculation process.

Two different time intervals will be defined

- Treatment cycle time (TC): Time between two consecutive processes of R/L/W.
- Control cycle time (CC): Time between two consecutive control actions.

The latter interval is an integral multiple of the former interval and it corresponds to the sample interval that PID's routines and other control routines must take into account.

Every TC the regulator enters the R/L/W process and every CC it sends a new control to the output. This control has been calculated in the timelags between the end and the begining of two consecutive R/L/W processes.

4. SOFTWARE ARCHITECTURE

Now we will describe the routines and data structures that support the processes above described. Let us begin with the noncyclic processes.

- Programming Monitor: This is the human interface that allows the regulator to be configured through the programming keyboard. In the configuration procedure the operator tells the Programming Monitor the structure he wants the regulator to have and the default values of its parameters. At the same time the Programming Monitor fills a DATA BASE where the structure is being implemented. The flexibility of the regulator is due to this data base and to a set of logical variables which are used to define the control logic of the regulator structure.

The Programming Monitor also permits to display the state of the main variables of the regulator and change their values once the regulation has started.

An important characteristic to notice is that in the configuration the operator defines the control law of the regulator. The way it is done is by telling the Programming Monitor the order in which some basic routines are linking together and the logical variables which govern their execution.

- Start-Up. The main function of this routine is to prepare the regulator to start the process of regulation. This means that it must transfer the values of the parameters of the regulator to their corresponding variables and give control to the routines that govern the cyclic process of regulation.

Let us continue with the routines and data structures associated with the cyclic processes.

- Programtheque: This is the name of the set of basic routines which are included in the regulator. They are divided into primary routines, say, the control routines and their associate bumpless routines which are used to define the control law, and the secondary routines which make the transference of variables among the primary routines used in the control law just defined.

- Table: This is a bidimensional array in which it is implemented the sequence of basic routines of the Programtheque which defines the control law of the regulator and the specific actions the regulator does in each operating mode. It is divided in four blocks, one for each mode. If we consider a single block each row contains the code of two routines and a logical variable whose state will decide which routine has to be executed.

- Analyzer: This routine analyses the block of the Table associated with the actual operating mode and obtain the code of the routine which has to be executed.

- Executive: This routine takes the code given by the Analyzer and execute the corresponding routine of the Programtheque.

Up to this point the routines associated with the process of control calculation has been considered.

- I/O Monitor: This routine carries out the process of R/L/W as well as the control of the synchronization between this process and the control calculation process.

The routine that is running continously is the Analyzer which is interrupted periodically by the I/O Monitor every CT. The Analyzer must accomplish the analysis of the Table in a CC so that the I/O Monitor be able to send a new control to the output in every CC.

The most remarkable characteristics of the regulator are the following:

1. The existence of a DATA BASE which allows the regulator to fit the peculiarities of a specific process. There are facilities so that a qualified operator be able to implement the desired structure on the data base.

2. The joining of all routines referring to the control action in a unique structure named Programtheque, and the definition of the control action for each operating mode in a bidimensional array called Table which is completely programmable. These two items make easier the change of the control law and the substitution of the actual control routines for new improved versions without modifying the global structure of the regulator.

3. The inclussion of an Adaptive operating mode.

In addition to the most usual industrial control algorithms (PID, PID+Gap, Cascade, Override, Ratio, Feedforward), the controller has an adaptive PID algorithm. When in the adaptive mode of operation there is an automatic tuning of the PID parameters. The basic scheme consists of a PID controller in parallel with an on-line pattern recognition algorithm which adjusts the controller tuning parameters.

The self-tuning algorithm is human like in its mode of operation. The internal circuitry measures the response of the system under control following a disturbance. The algorithm uses the registered response information to compute the news control coefficients. The method followed by the algorithm is in some aspects similar to that of (4).

CONCLUSION

The computing power of microprocessors makes that the general purpose control systems designed for use on large number of processes systems be increasing their complexity in this paper. An industrial programmable adaptive controller has been considered.

Their functional and software structure has been discussed. The controller has four operation modes: manual, automatic, computer, and adaptive. The controller has been designed to be compatible with most PID controllers existing nowadays; it may be configured through a programming keyboard. The analog signals and algorithms are given a fitted configuration to get the structure the operator wants the controller to have. An adaptive mode of operation has been included. In this mode there is an automatic tuning of the PID parameters.

REFERENCES

(1) Comparing the Relative Complexities of Programming Process Controllers. Kompas, E.J. and Morris, H.M., Control Engineering, vol. 28, nº 7, pp. 75-78, 1981

(2) Direct Digital Process Control: Practice and Algorithms for Microprocessor Application. D.M. Auslander, Y. Takahashi, and M. Tomizuka Proc. IEEE, vol. 66, No. 2, pp. 199-208, 1978.

(3) Implementation and Application of Microprocessor Based Self-tuners. D.W.Clarke, P.J. Gawthrop. Automatica, vol. 17, pp. 233-244, 1981.

(4) Self-Tuning PID Controller Uses Pattern Recognition Approach. T.W. Kraus, T.J. Myron. Control Engineering, June 1984, pp. 106-111.

(5) A Method for Auto-tuning of PID Control Parameters. Y. Nishikawa, N. Sannomiya, and H. Tanaka. Automatica, vol. 20, pp. 321-332, 1984.

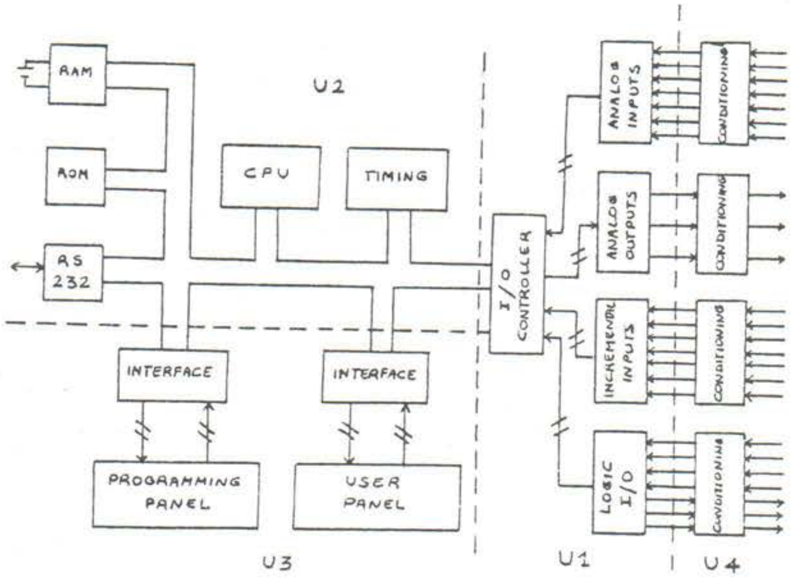

Fig.1: Hardware structure of general purpose control systems