Scientific Journals
Maritime University of Szczecin

Zeszyty Naukowe
Akademia Morska w Szczecinie

2013, 36(108) z. 2 pp. 75–79
ISSN 1733-8670

2013, 36(108) z. 2 s. 75–79

Implementation of typical models of control plants with use of programmable logical controllers

Leszek Kaszycki

Maritime University of Szczecin, Institut of Marine Electrical Engineering and Vessel Automation
70-500 Szczecin, ul. Wały Chrobrego 1–2, e-mail: l.kaszycki@am.szczecin.pl

Key words: programmable controller, continuous model, discrete model, step response, transfer function

Abstract
The paper demonstrates the possibility of synthesis of basic models of control plants with use of programmable controller. Selected models of plants were synthesized basing on standard functional blocks and arithmetic of integer type. Also was presented an application of Versa Max type controller in synthesis of discrete models of oscillatory and inertial blocks.

Introduction

Software used in contemporary programmable logical controllers (PLC) allows to easy generation of algorithms (e.g. in a ladder program form) widely applied in control engineering. Available standard function can be as follows: contacts, relays, counters, timers, functional blocks implementing mathematical operations, or PID type controllers [1].

Due this reason construction of classic control systems (e.g. continuous, two-point or sequential control systems) with use PLC is relatively easy.

In case of modern algorithms (e.g. predictive), or systems with inner model [2, 3], synthesis of ladder diagram with use of standard block available in PLC is not so obvious.

Main problem of implementation is model of control plant, especially with use of simplest PLC of micro class.

The following paper shows way of implementation of typical models of control plants with use of GE-Fanuc 90–30 Micro, as well as Versa Max Micro and Proficy Machine Edition software.

Basic linear models of plants

Due to properties of step response, linear control plants an be divided into two groups: static, with limited step response and astatic, with unlimited step response values (Fig. 1).

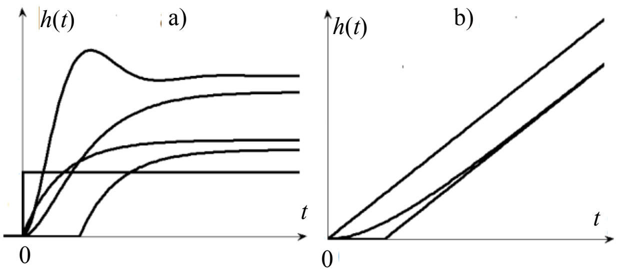

Fig. 1. Step responses of a) static plants, b) astatic plants

Typical models of control plant are as follows:

– first order inertia block with delay, described by transfer function:

$$G(s) = \frac{k}{Ts+1} \cdot \exp(-\tau \cdot s) \tag{1}$$

– higher order inertia block with transfer function:

$$G(s) = \frac{k}{(T_1 s+1) \cdot (T_2 s+1) \cdot \ldots \cdot (T_n s+1)} \tag{2}$$

where: k – gain factor; T, T_1,... T_n – time constants; τ – delay time.

– oscillatory block, described by transfer function:

$$G(s) = \frac{k \cdot \omega_0^2}{s^2 + 2\beta s + \omega_0^2} \tag{3}$$

where: k – gain factor; ω_0 – frequency of undamped oscillations; β – damping factor.

Models of astatic plants are as follows:

– integrator with delay, described by transfer function:

$$G(s) = \frac{1}{T_c \cdot s} \cdot \exp(-\tau \cdot s) \tag{4}$$

where: T_c – integration time; τ – delay time.

– serial connection of integrator with first order inertia block:

$$G(s) = \frac{1}{T_1 \cdot s} \cdot \frac{1}{T_2 s + 1} \tag{5}$$

Presented above models are typical, and do not cover a whole range of other, used in description of control processes.

Synthesis of selected models of control plants

It is easy to find, that each of models described with formulae (1)–(5) is serial connection of basic elements, such as below:

– first order inertia block, described by transfer function:

$$G(s) = \frac{k}{Ts + 1} \tag{6}$$

– ideal integrating block:

$$G(s) = \frac{1}{T_c \cdot s} \tag{7}$$

– proportional block:

$$G(s) = k \tag{8}$$

– delay block:

$$G(s) = \exp(-\tau \cdot s) \tag{9}$$

In order to model plants (1)–(5), it is enough to build simpler models, given by (6)–(9), and to take in account their serial connections.

The following functional blocks were selected to software implementation with use Proficy Machine Edition:

- PID-IND – controller of PID type with independent terms;
- SHFR WORD – shift register;
- MUL – arithmetic multiplicator.

Functional block PID-IND is presented in figure 2.

Algorithm of PID-IND is described by formula:

$$y(t) = k_r \cdot e(t) + \frac{1}{T_c}\int_0^t e(t)\,\mathrm{d}t + T_d \cdot \frac{\mathrm{d}e(t)}{\mathrm{d}t} \tag{10}$$

where error signal $e(t)$ and controlled signal $y(t)$ satisfy the following relations:

$$e(t) = \mathrm{SP} - \mathrm{PV}, \quad y(t) = \mathrm{CV}$$

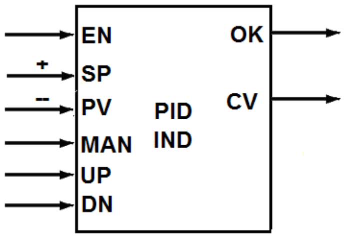

Fig. 2. Functional block PID-IND: EN – enabling signal, PV – controlled signal, CV – controlling signal, SP – set-point value, MAN – selection manual / automatic control, UP, DN – increase or decrease in automatic mode, OK – confirmation of properly execution of PID algorithm in given controller cycle

Coefficients k_r, T_c and T_d are controller gain factor, integration time and derivation time respectively, and can be tuned within wide ranges in programmatically way.

Relations (6) and (10) suggest the possibility of unusual application of PID-IND functional block for the synthesis of first order inertia block. This implementation is shown in figure 3.

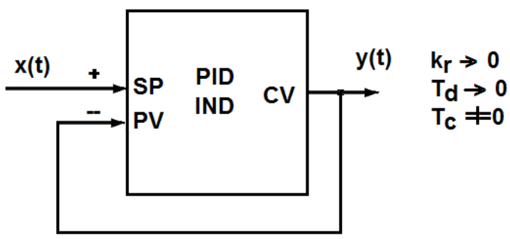

Fig. 3. Synthesis of first order inertia block

Description of this system in time domain is given by formula:

$$y(t) = \frac{1}{T_c} \cdot \int_0^t [x(t) - y(t)]\mathrm{d}t \tag{11}$$

Denoting $x(t)$ as input signal, and $y(t)$ as output signal and applying to (11) Laplace transformation, the following transfer function is obtained as result:

$$G(s) = \frac{k}{T_c s + 1}$$

Feedback loop shown in figure 3 should be interpreted as realised programmatically.

Program written for GE-Fanuc 90–30 Micro PLC according to structure from figure 3 was tested. The example of recorded step response, for amplitude of input equal to 10 units, and T_c = 10 s is shown in figure 4.

Integrating block described by (7) can be easy obtained wit use the same PID-IND functional block. Settings of controller should be as shown in figure 5.

The delay block described with use of transfer function (8) can be implemented in easy way in PLC as subprogram generating queue memory, applied in functional block SHFR-WORD. An example of implementation with delay time $\tau = 60$ s is shown in figure 6.

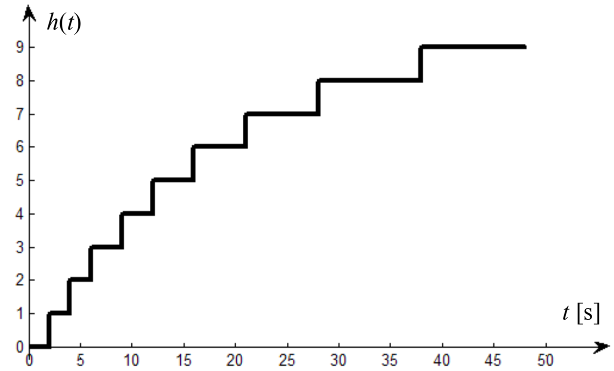

Fig. 4. Step response of model with transfer function (12)

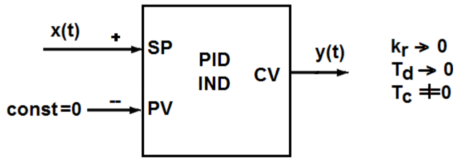

Fig. 5. Model of integrating block

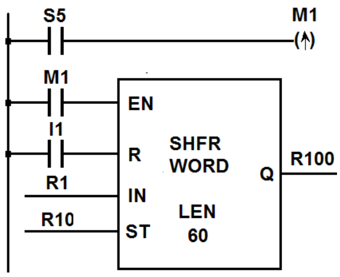

Fig. 6. Subprogram with delay element ($\tau = 60$ s)

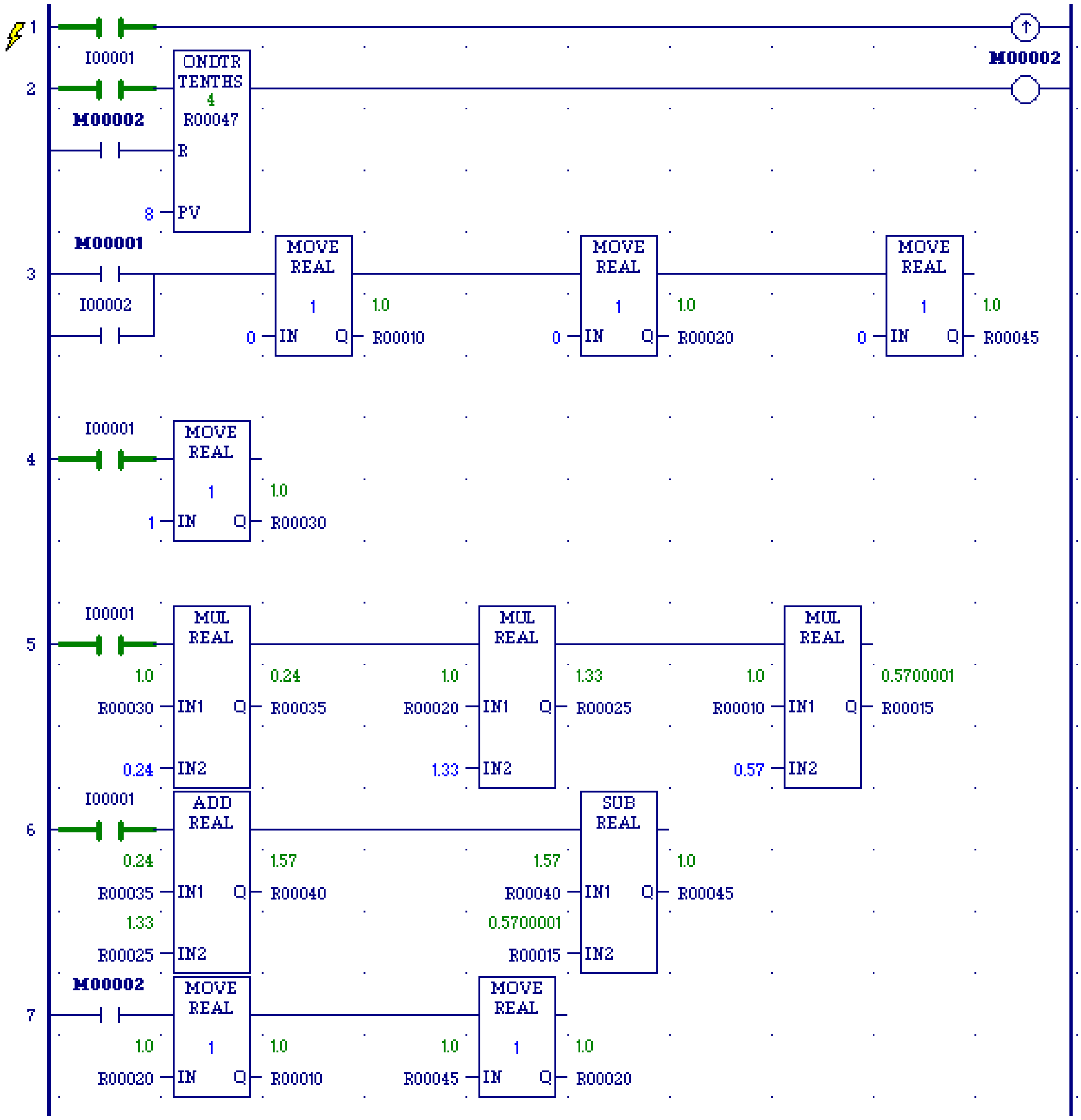

Fig. 7. Subprogram modelling oscillatory block

In this case queue memory was built with 60 registers addressed beginning from R10 address. Value of input signal (number stored in R1 register) appears at the output (register R100) after 60 seconds. Shifting of binary words in queue memory is executed with frequency of 1 Hz. Controlling signal is generated by M1 relay.

Synthesis of oscillatory block of with use single standard functional block (e.g. PID) is not possible, instead this, the discrete model of block should be built.

Differential equation, describing oscillatory block has a form:

$$\frac{1}{\omega_0^2}\cdot\frac{d^2 y(t)}{dt^2}+\frac{2\beta}{\omega_0}\cdot\frac{d\,y(t)}{dt}+y(t)=k\cdot x(t) \quad (12)$$

Taking in regard that backward approximations of derivatives [3] are equal to:

$$\frac{d\,y(t)}{dt}\approx\frac{y_n-y_{n-1}}{h} \quad (13)$$

$$\frac{d^2 y(t)}{dt^2}\approx\frac{y_n-2y_{n-1}+y_{n-2}}{h^2} \quad (14)$$

where h denotes sampling period.

Using (13) and (14) equation (12) can be transformed into discrete form:

$$y_n=A\cdot x_n+B\cdot y_{n-1}-C\cdot y_{n-2} \quad (15)$$

where:

$$C=\frac{1}{1+2\beta\omega_0 h+\omega_0^2 h^2} \quad (16)$$

$$A=C\cdot k\omega_0^2 h^2 \quad (17)$$

$$B=C\cdot 2(1+\beta\omega_0 h) \quad (18)$$

Subprogram (prepared for PLC of Versa Max Micro type) with discrete model of oscillatory block is shown in figure 7. Register R30 contains value of input variable x, value of output variable y is stored in register R45.

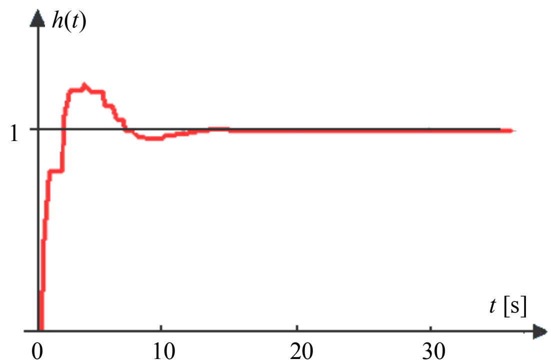

Fig. 8. Obtained step response of oscillatory block

For assumed values of parameters: $h = 1$ s, $\beta = 0.25$, $\omega_0 = 0.65$ rad/s, $k = 1$ were obtained the following values: $A = 0.24$, $B = 1.33$, $C = 0.57$. Course of step response of model of oscillatory block is shown in figure 8.

In analogous way was programmed discrete version of model of first order inertia block, described with use of differential equation:

$$T\frac{d\,y(t)}{dt}+y(t)=k\cdot x(t) \quad (19)$$

Obtained discrete model has a form:

$$y_n=A\cdot x_n+B\cdot y_{n-1} \quad (20)$$

where:

$$A=k\frac{h}{T};\,B=1-\frac{h}{T} \quad (21)$$

Assumed values of parameters are as follows: $k = 1$, $h = 1$ s, $T = 10$ s. Program destined to modelling of step response is shown in figure 9, and course of obtained step response in figure 10.

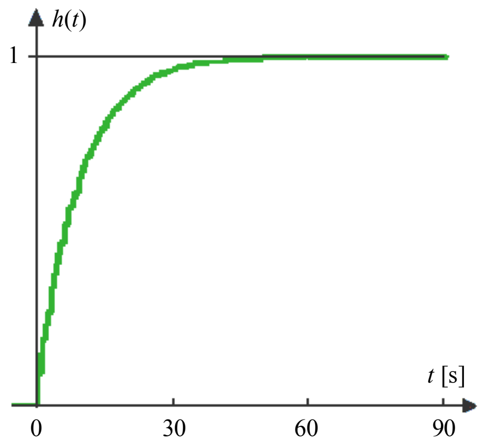

Fig. 10. Step response of first order inertia block

Conclusions

Theoretical analysis, as well as results of simulative tests confirmed possibility of synthesis of basic models of control plants with use of programmable logical controllers (PLC).

Majority of mentioned models can be easy implemented even with use very simple PLC of Micro class with use one of standard functional block and arithmetic of integer type. Application to this aim functional block of PID-IND type needs use of about 30 registers of PLC [1]. In case of PLC with arithmetic of floating-point type, discrete models are simple in programming and engage significant less of PLC memory in comparison with PLC equipped with arithmetic of integer type.

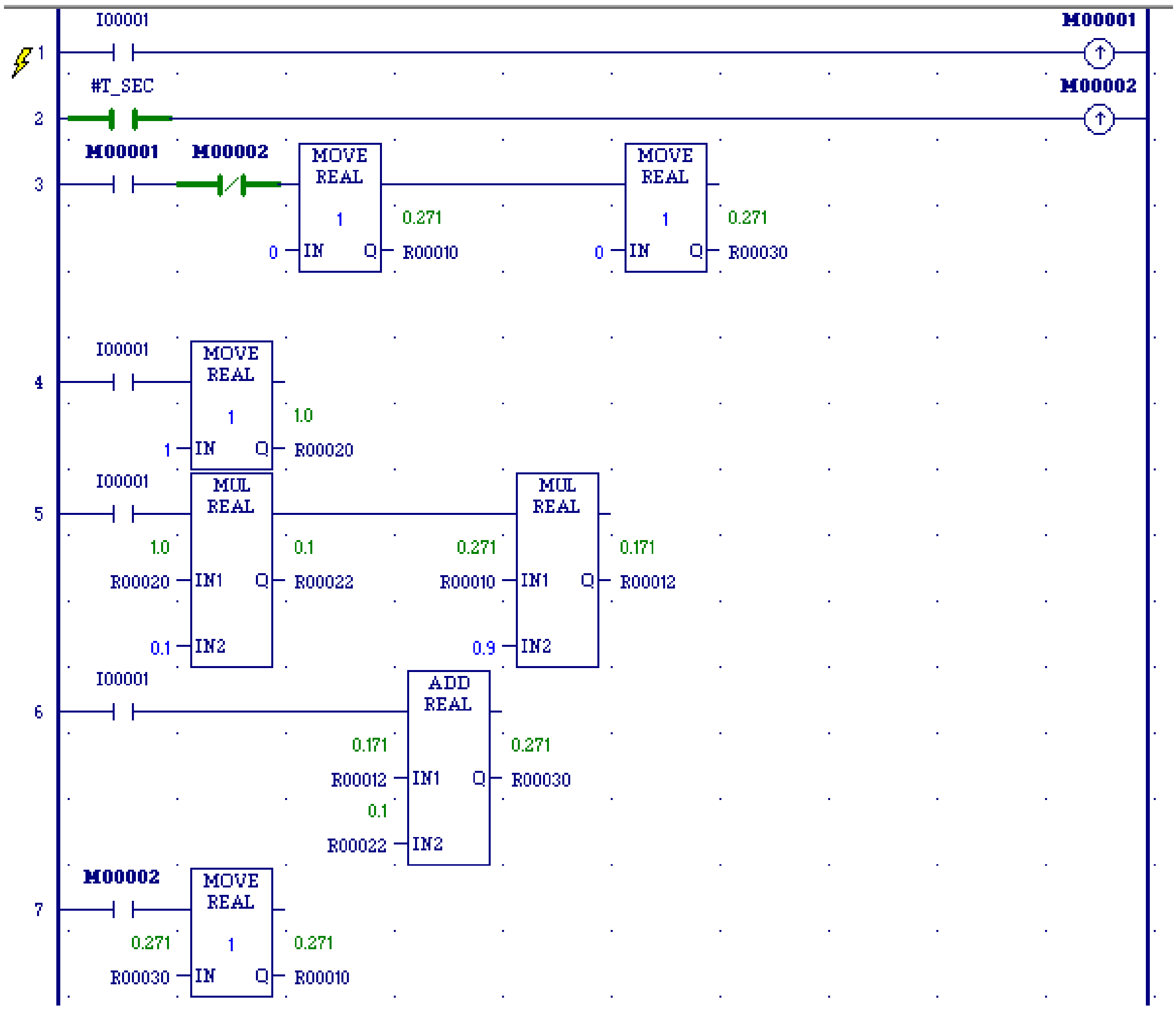

Fig. 9. Subprogram modelling first order inertia block (Register R20 contains value of input variable *x*, value of output variable *y* is stored in register R30)

References

1. GE Fanuc Automation Series 90-30/20 Micro Programmable Controllers. Reference manual, GFK-04676, 1996.
2. Skoczowski S.: Odporny układ regulacji z wykorzystaniem modelu obiektu. Pomiary, Automatyka, Kontrola 9, 1999, 2–4.
3. Grega W.: Metody i algorytmy sterowania cyfrowego w układach scentralizowanych i rozproszonych. Uczelniane Wydawnictwo Naukowo-Dydaktyczne AGH w Krakowie, Kraków 2004.