

dr inż. Piotr Szulewski  
 Instytut Techniki Wytwarzania, Politechnika Warszawska  
 mgr inż. Arkadiusz Welk  
 Centrum Badawczo-Konstrukcyjne Obrabiarek, Pruszków

## PROCESOROWY UKŁAD POZYCJONOWANIA WAŁU SILNIKA

*Powszechnie występująca tendencja związana z wykorzystywaniem w obszarze automatyki przemysłowej (do sterowania i nadzoru) systemów mikroprocesorowych wymusza potrzebę badań i testów tego typu urządzeń. Przedstawiony w pracy mikroprocesorowy układ pozycjonowania wału silnika oparty jest o algorytm dyskretnego regulatora PID. Cechą odróżniającą go od spotykanych na rynku rozwiązań jest wykorzystanie prostego 8-bitowego mikrokontrolera o niewielkiej mocy obliczeniowej i otwarta struktura oprogramowania umożliwiająca swobodne zmiany parametrów a także komunikację z dedykowanym oprogramowaniem monitorującym umieszczonym na komputerze klasy PC.*

### MICROPROCESSOR BASED DRIVE SHAFT POSITION CONTROLLER

*Thanks to the microelectronic technique rapid growth it is possible to use the popular microprocessor and it's components as universal and powerful controllers. Referring to the common idea of microprocessor based controller the conception of easy and cheap structure for drive shaft position control is presented in this paper. The digital PID algorithm (slightly simplified) was implemented into controller. The communication with PC for effective data acquisition was established. Some basic theory was slightly illuminated as well.*

## 1. CYFROWY REGULATOR PID

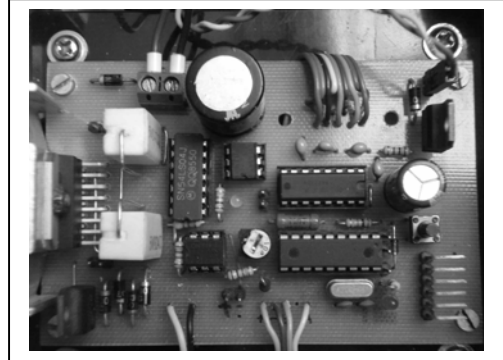
Zasady działania tradycyjnego regulatora PID są łatwe do zrozumienia, a podstawowa wiedza na jego temat jest ogólnie dostępna. Algorytm PID dzięki możliwości zastosowania techniki mikroprocesorowej, głównie za sprawą środowisk akademickich, doczekał się wielu swoich odmian-mutacji. Jednym z takich nietypowych rozwiązań jest na przykład wprowadzenie nieliniowości – np. regulator, który w torze P reaguje na kwadrat sygnału błędu [Brzózka, 04]. Korzystając z osiągnięć współczesnej techniki cyfrowej możliwe jest także zaimplementowanie algorytmu sterowania PID do prostego systemu mikroprocesorowego. W takiej postaci mamy do czynienia z regulatorem dyskretnym, który charakteryzuje się cyklicznością pracy – sygnał przetwarzany jest w kolejnych próbkach. Układem sterującym jest zaprogramowany mikrokontroler – w omawianym tu przypadku steruje on pozycjonowaniem kątowym wału silnika. Ze względu na dyskretny charakter pracy mikrokontrolera (i związany z tym czas wykonywania programu – czyli czas próbkowania regulatora), do symulacji komputerowej mikroprocesorowego układu pozycjonowania wału silnika przyjęto model dyskretnego regulatora PID. Prawo regulacji dyskretnego regulatora PID jest przedstawione za pomocą wzoru:

$$u(n) = \underbrace{Kp \cdot e(n)}_P + \underbrace{Ki \cdot \frac{T_s}{T_i} \cdot \sum_{i=0}^n e(n)}_I + \underbrace{Kd \cdot \frac{T_d}{T_s} [e(n) - e(n-1)]}_D \quad (1)$$

gdzie odpowiednio oznaczono:  $e(n)$  – uchyb w  $n$ -tym okresie próbkowania,  $K_p$  – wzmacnienie członu proporcjonalnego,  $K_i$  – wzmacnienie członu całkującego,  $K_d$  – wzmacnienie członu różniczkującego,  $T_i$  – czas całkowania,  $T_d$  – czas różniczkowania,  $T_s$  – czas próbkowania.

Sygnałem wejściowym regulatora jest uchyb pozycjonowania wału silnika  $e(n)$ . Uchyb został zdefiniowany w następujący sposób:  $e(n) = \text{pozycja wału wymagana} - \text{pozycja wału rzeczywista}$ . Wielkość opisująca *pozycję wału* silnika wyrażona została liczbą działek enkodera. Sygnał wejściowy regulatora cyfrowego  $e(n)$  przetwarzany jest w każdym cyklu pracy programu. Czas trwania tego cyklu określa czas próbkowania  $T_s$ . Widok płytki regulatora cyfrowego jest zamieszczony na rys. 1.

W przypadku tego układu sterowania zastosowano regulator PID, który umożliwia niezależny dobór wzmocnienia dla każdego członów – współczynniki  $K_p$ ,  $K_i$ ,  $K_d$ . Poprzez wyzerowanie wybranych współczynników wzmocnienia, można wyłączyć dany człon i w wyniku otrzymać regulator: P, I, PI lub PD. Ze względu na planowane wykorzystanie układów cyfrowych o małej mocy obliczeniowej zastosowano uproszczenia algorytmu pozwalające na zmniejszenie liczby operacji arytmetycznych realizowanych przez mikroprocesor.



Rys. 1. Mikroprocesorowy regulator PID

W przedstawionym regulatorze cyfrowym całkowanie zastąpiono sumą. Dodawanie wartości uchybu do akumulatora całkowania wykonuje się w każdym cyklu pracy regulatora. W ten sposób przy małym uchybie o długim czasie trwania, regulator może wygenerować sygnał, o nasyceniu wystarczającym do zainicjowania reakcji układu sterowanego. Akumulator całkowania może przechowywać liczby od -127 do +127 z dokładnością 16 bitów – dzięki temu limitowane jest nieograniczone narastanie sygnału spowodowane całkowaniem. Uchyb zanim zostanie dodany do akumulatora całkowania, jest skalowany przez stosunek czasu próbkowania  $T_s$  do czasu całkowania  $T_i$ . Przed dodaniem zawartości akumulatora całkowania do sumy regulatora, akumulator jest skalowany według współczynnika wzmocnienia  $K_i$  członu całkującego.

W omawianym regulatorze cyfrowym różniczkowanie zastąpiono różnicą. Wylisanie różnicy uchybu wykonywane jest w każdym cyklu próbkowania regulatora. Różnica liczona jest między aktualnym uchybem  $e(n)$  a uchybem, który istniał w poprzednim okresie próbkowania  $e(n-1)$  – jest to różnica regresywna (wsteczna) pierwszego rzędu. W ten sposób wyliczona różnica mnożona jest przez stosunek czasu różniczkowania  $T_d$  do czasu próbkowania  $T_s$ , a następnie skalowana przez współczynnik wzmocnienia członu różniczkującego  $K_d$ .

Możliwe do uzyskania wartości nastaw czasów regulatora oraz zależności między nimi przedstawia tab. 1. Czas próbkowania regulatora może być swobodnie zmieniany w zakresie od 0,2 ms do 104,9 ms ze skokiem 1,6  $\mu\text{s}$ . Precyzja nastawy czasu próbkowania wynika z częstotliwości impulsów zliczanych przez 16-bitowy licznik stabilizujący czas próbkowania. Częstotliwość pracy tego licznika jest równa 1/32 częstotliwości  $f$  taktującej mikrokontroler. Czasowi próbkowania równemu 200  $\mu\text{s}$  odpowiada nastawa licznika  $T_{s16}$  równa 125.

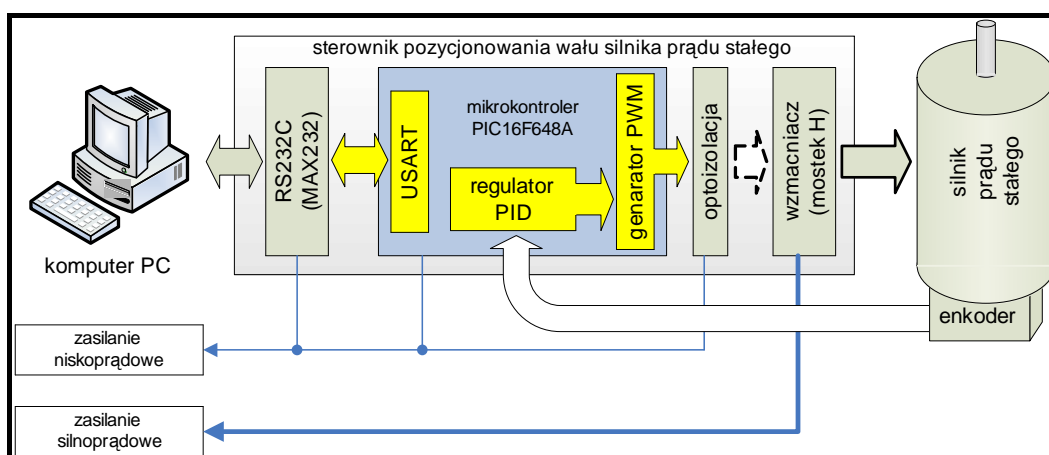
**Tab. 1.** Zakres nastaw czasowych regulatora (Welk, 06)

Opis	symbol lub zależność	wartości nastaw	
częstotliwość zegara taktującego	F	20 MHz	
<b>CZAS PRÓBKOWANIA</b>	$T_{S16}$	min = 125	max = 65536 [16 bit]
precyzja nastawy	32 / f	1,6 $\mu$ s	
zakres nastaw	$1,6 \cdot T_{S16}$	200 $\mu$ s	104,9 ms
częstotliwość próbkowania	$(1,6 \cdot T_{S16})^{-1}$	5 kHz	9,53 Hz
<b>CZAS CAŁKOWANIA</b>	$T_{i8}$	min = 1	max = 255 [8 bit]
rozdzielczość nastawy	$T_i = T_{i8} \cdot T_s$	200 $\mu$ s	104,9 ms
zakres nastaw (dla $T_s=200 \mu$ s)		0,2 ms	51 ms
zakres nastaw (dla $T_s=104,9$ ms)		0,1049 S	26,75 S
<b>CZAS RÓŻNICZKOWANIA</b>	$T_{d8}$	min = 1	max = 255 [8 bit]
rozdzielczość nastawy	$T_d = T_{d8} \cdot T_s$	200 $\mu$ s	104,9 ms
zakres nastaw (dla $T_s=200 \mu$ s)		0,2 ms	51 ms
zakres nastaw (dla $T_s=104,9$ ms)		0,1049 s	26,75 s

Czas całkowania  $T_i$  i czas różniczkowania  $T_d$  regulatora są wielokrotnością czasu próbkowania  $T_s$ . Wielokrotność tę określają 8-bitowe rejestry  $T_{i8}$  i  $T_{d8}$ , przy czym uzyskane w ten sposób czasy  $T_i$  i  $T_d$  są zawsze większe niż  $T_s$ .

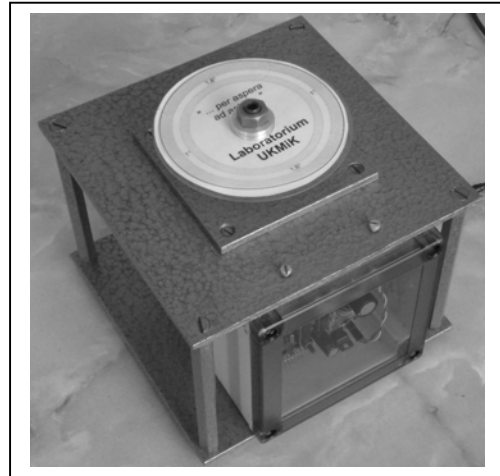
## 2. POZYCJONOWANIE WAŁU SILNIKA

Opisywany w tej pracy mikroprocesorowy układ pozycjonowania wału silnika prądu stałego oparty jest właśnie na algorytmie dyskretnego regulatora PID. Opisany, cyfrowy algorytm regulacji PID ze względu na swoją niewielką złożoność, umożliwia implementację w prostych 8-bitowych mikrokontrolerach o niewielkiej (w stosunku do współczesnych komputerów) mocy obliczeniowej, co czyni to rozwiązanie niezwykle tanim. Mikrokontroler z zaimplementowanym algorytmem regulatora PID musi być wyposażony w odpowiednie urządzenia dopasowujące sygnały wejścia/wyjścia (np. układ silnopiętowy w przypadku sterowania urządzeniami pobierającymi znaczna moc), oraz kod do ich obsługi, aby umożliwić współpracę z wybranym obiektem regulacji. Schemat ideowy przedstawionego sterownika (mikrokontroler + układy dopasowujące) znajduje się na rys. 2.

**Rys. 2.** Schemat blokowy układu pozycjonowania kąowego wału silnika prądu stałego (Welk, 06)

Sterownik pozycjonowania wału silnika prądu stałego zbudowano w technice mikroprocesorowej w oparciu o 8-bitowy mikrokontroler firmy Microchip o symbolu PIC16F648A.

Do zmiany parametrów pracy sterownika i nastaw regulatora PID służy dedykowane, samodzielnie opracowane, oprogramowanie zainstalowane w komputerze PC, pracujące pod kontrolą systemu operacyjnego Windows (w wersji 32-bitowej). Oprogramowanie to komunikuje się z mikrokontrolerem poprzez sprzętowy port szeregowy USART (ang. uniwersal synchronous asynchronous receiver transmitter, uniwersalny układ nadawczo-odbiorczy do transmisji synchronicznej i asynchronicznej) mikrokontrolera. Jako konwerter napięć portu szeregowego mikrokontrolera do standardu RS-232C zastosowano układ MAX232 (linie transmisji danych Tx – nadawcza i Rx – odbiorcza) układu USART mikrokontrolera pracują w zakresie napięć od 0 V do 5 V. Standard RS-232C wymaga zakresu napięć linii Tx/Rx od -12 V do 12 V). Transmisja danych odbywa się za pośrednictwem specjalnego protokołu komunikacji.



Rys. 3. Układ cyfrowego pozycjonowania wału silnika

Regulator PID zaimplementowany w mikrokontrolerze przystosowano do współpracy z enkoderem obrotowo impulsowym (urządzenie wejścia) sprzężonym mechanicznie z wałem silnika prądu stałego. Silnik zasilany jest przez wzmacniacz, którym jest układ silnoprądowy L298, pracujący w konfiguracji jako mostek typu H (urządzenie wyjścia). L298 sterowany jest sygnałem o zmiennej szerokości wypełnienia z generatora PWM (PWM – ang. pulse width modulation, modulacja szerokości impulsu) wbudowanego w mikrokontroler. Rozdzielczość wejścia sterującego wypełnieniem sygnału PWM wynosi także 8 bitów.

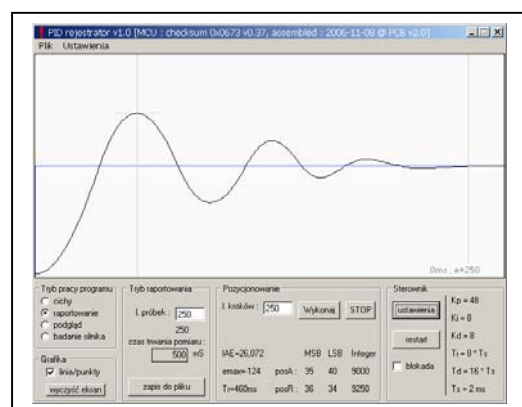
Aby zapobiec przedostawaniu się zakłóceń elektromagnetycznych generowanych w silniku (efekt przełączania cewek wirnika przez komutator) do części mikroprocesorowej układu, zastosowano oddzielne obwody zasilające część mikroprocesorową i część silnoprądową, a sygnały sterujące mostkiem H doprowadzono z mikrokontrolera poprzez optoizolację (transoptory CNY17). Widok kompletnego stanowiska (regulator + silnik) jest przedstawiony na rys. 3.

### 3. APLIKACJA BADAWCZA

Dla umożliwienia monitorowania (śledzenia) stanu pracy regulatora, stworzono w środowisku języka Visual Basic 6.0 aplikację badawczą pod nazwą „PID\_rejestrator ver. 0.1”, ułatwiającą współpracę między mikroprocesorowym sterownikiem pozycjonowania, a komputerem PC. Wygląd okna głównego aplikacji badawczej znajduje się na rys. 4.

Aplikacja to pozwala na:

- wysyłanie rozkazów do sterownika pozycjonowania dot. zmiany pozycji wału silnika



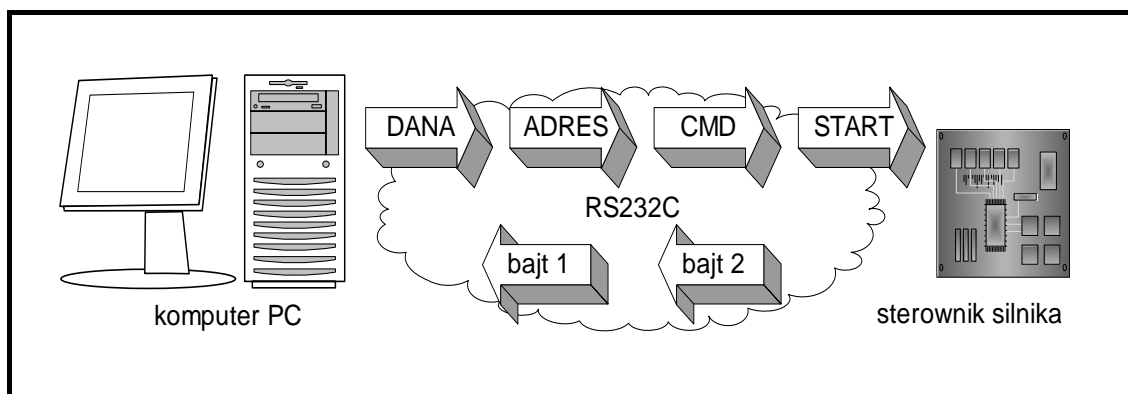
Rys. 4. Okno główne aplikacji badawczej

- zbieranie i analizę danych uchybu podczas wykonywania pozycjonowania
- zmianę parametrów pracy mikroprocesorowego sterownika pozycjonowania
- zmianę nastaw regulatora PID, sterującego procesem pozycjonowania
- pomiar stałych czasowych  $T_o$  i  $T_{io}$  silnika.

Transmisja danych przebiega w obu kierunkach za pośrednictwem sprzętowego portu szeregowego wbudowanego w układ mikrokontrolera, pracującego w standardzie RS-232C. Dane charakteryzujące transmisję szeregową pomiędzy sterownikiem a komputerem PC to :

- prędkość transmisji 115,2 Kbps
- 8-bitowa ramka danych
- brak bitu parzystości
- 1 bit stopu.

Obsługą programową transmisji danych od strony mikrokontrolera zajmuje się specjalny protokół komunikacji, umożliwiający pełny dostęp (zapis i odczyt) do całej pamięci operacyjnej (RAM) oraz nieulotnej pamięci danych (EEPROM) mikrokontrolera. Poprzez modyfikację komórek pamięci RAM i EEPROM można bezpośrednio wpływać na pracę mikrokontrolera i zaimplementowanego w nim regulatora PID. Komunikacja odbywa się na zasadzie pytanie-odpowieź, gdzie układem nadrzędnym (master), inicjującym transmisję jest system zewnętrzny (np. komputer PC), a układem podrzędnym (slave) jest mikrokontroler. Przepływ danych podczas transmisji przedstawia rys. 5.



Rys. 5. Model komunikacji regulatora z komputerem PC

Obsługa transmisji danych w sterowniku odbywa się tylko poprzez przerwanie. Komunikacja może rozpocząć się tylko wtedy, gdy z systemu nadrzędnego do mikrokontrolera zostanie wysłany odpowiedni bajt START, któremu odpowiada liczba 255. Jeżeli pierwszy odebrany przez sterownik bajt nie był bajtem startu transmisji (nie jest liczbą 255), to układ opuszcza kod odpowiedzialny za obsługę komunikacji i następnie kończy obsługę przerwania na to zdarzenie. Pełna transmisja danych realizowana jest tylko wtedy, gdy odebrany przez mikrokontroler pierwszy bajt jest bajtem startu, czyli jest to liczba równa 255. Wówczas uruchamiana jest procedura odbioru pozostałych bajtów – w tym przypadku czas mikrokontrolera poświęcany jest tylko na transmisję danych. Bajt drugi (CMD) wysyłany przez układ master, zawiera rozkaz dla mikrokontrolera. Bajt trzeci (ADRES) i czwarty (DANA), są argumentami dla wysłanego rozkazu. Po pobraniu kompletnej czterobajtowej ramki pytania i wykonaniu żadanego rozkazu, mikrokontroler wysyła odpowiedź (wiadomość zwrotną) do układu master. Ramka odpowiedzi składa się z dwóch bajtów, których znaczenie jest różne w zależności od 5. bitu odebranego rozkazu. Istnieje możliwość zmiany prędkości

transmisji szeregowej, jednak w tym przypadku należy najpierw odpowiednio skonfigurować układ sterownika mikroprocesorowego. Bezpośredni dostęp do wszystkich komórek pamięci sterownika umożliwia program PICCOM – dostarczany przez producenta mikrokontrolera.

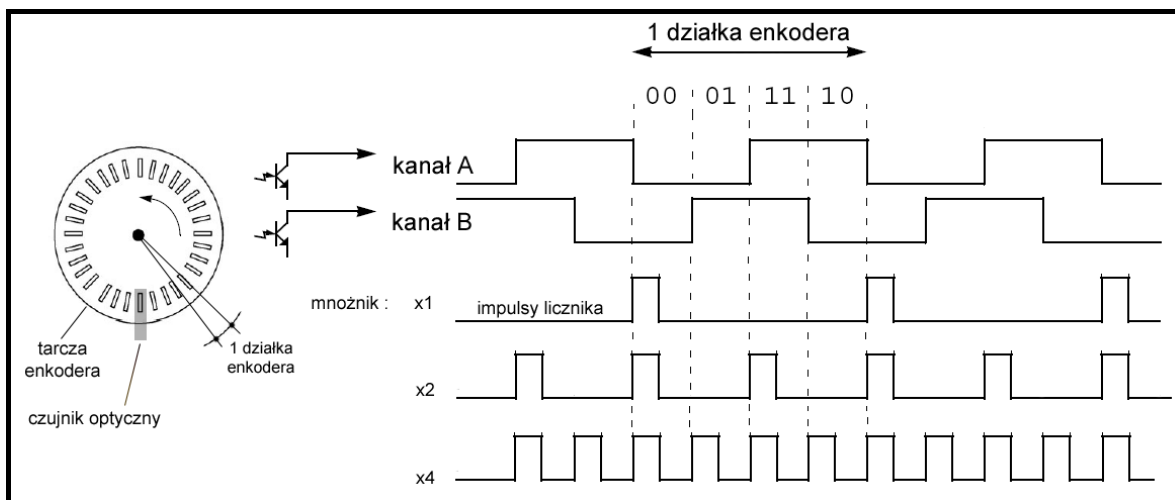
Aplikacja może pracować w kilku trybach, różniących się sposobem funkcjonowania:

<b>Cichy</b>	Aplikacja nie prowadzi rejestracji przebiegu uchybu podczas pozycjonowania wału. Tryb ten umożliwia tylko wysyłanie z komputera PC do sterownika, rozkazów zmiany położenia kąтового wału o wskazaną przez użytkownika liczbę kroków, wyrażoną w ilości działek enkodera. Regulator PID reaguje w czasie rzeczywistym i odpowiednio steruje procesem, w celu zminimalizowania uchybu.
<b>Raportowanie</b>	Rejestrowany jest przebieg położenia wału w czasie wykonywania pozycjonowania. Przed wysłaniem rozkazu zmiany położenia wału, aplikacja wysyła do mikrokontrolera informację o ilości próbek, które mikrokontroler musi zgłosić do komputera. Komputer w czasie wykonywania pozycjonowania prowadzi nasłuch portu, do którego został przyłączony sterownik pozycjonowania i rejestruje wszystkie odebrane dane.
<b>Podgląd</b>	Jest mniej precyzyjnym sposobem na rejestrację przebiegu zmiany położenia wału w czasie niż tryb raportowania. Mniejsza dokładność wynika z faktu, że to aplikacja decyduje o tym, kiedy mikrokontroler ma wysłać kolejną próbkę – komputer nie prowadzi nasłuchiwanie portu (jak w przypadku <i>raportowania</i> ), lecz ciągle wysyła do mikrokontrolera żądania odczytu aktualnego położenia wału. Jedno wysłane przez aplikację żądanie odpowiada jednej próbce wysłanej z mikrokontrolera. Umożliwia rysowanie w czasie rzeczywistym wykresu przebiegu zmiany położenia wału. Wadą tego trybu jest: duży odstęp czasu między kolejnymi próbkami (rzędu 100ms).
<b>Badanie silnika</b>	Aplikacja realizuje pomiar stałych czasowych charakteryzujących silnik podłączony do sterownika. Uzyskiwanie w tym trybie informacji o przebiegu położeniu wału, odbywa się na tej samej zasadzie co w trybie raportowania. Użytkownik określa ilość próbek, jaka ma być zgłoszona przez mikrokontroler. Czas próbkowania, przy którym zostanie przeprowadzone doświadczenie, może być inny niż $T_s$ , lecz musi zawierać się w zakresie od 0,4 ms do 10 ms. W przypadku doboru czasu próbkowania do procesu pomiaru stałych czasowych jest zasada, że im krótszy czas próbkowania, tym dokładniejszy wynik. W tym trybie pracy, aplikacja rejestruje odpowiedź silnika na wymuszenie skokowe. Odpowiedzią silnika jest zmiana położenia wału. Wymuszeniem jest skok napięcia zasilającego silnik od zera do wartości maksymalnej $U_n$ lub $-U_n$ oferowanej przez mostek H. Użytkownik określa, jaki ma być kierunek polaryzacji silnika podczas przeprowadzonego doświadczenia.

#### 4. WSPÓŁPRACA Z ENKODEREM

Napisane oprogramowanie sterujące oraz zastosowany mikrokontroler umożliwiają zwiększenie rozdzielczości pomiaru z przetwornika obrotowo/impulsowego. W przypadku enkodera można ustawić mnożnik odpowiadający za rozdzielczość odczytu na  $1\times$ ,  $2\times$ , lub  $4\times$ . Rozdzielczość odczytu enkodera przez sterownik związana jest ze sposobem reagowania sterownika na impulsy generowane przez enkoder (rys. 6.). Sterownik może reagować na jeden typ zbocza (np. tylko narastające) na jednym kanale enkodera i wtedy mnożnik wynosi  $1\times$ . Przy tym mnożniku uzyskana rozdzielczość odczytu jest równa tej, którą podaje producent enkodera w karcie katalogowej. Gdy zostanie ustawiony mnożnik na  $2\times$ , wtedy sterownik reaguje na zbocze narastające i opadające na jednym kanale enkodera, a uzyskana w ten sposób rzeczywista rozdzielczość odczytu będzie równa dwukrotnej rozdzielczości enkodera, którą zapewnia producent. W przypadku mnożnika  $4\times$  sterownik reaguje na każdy typ zbocza sygnału odebranego z każdego z dwóch kanałów enkodera obrotowo

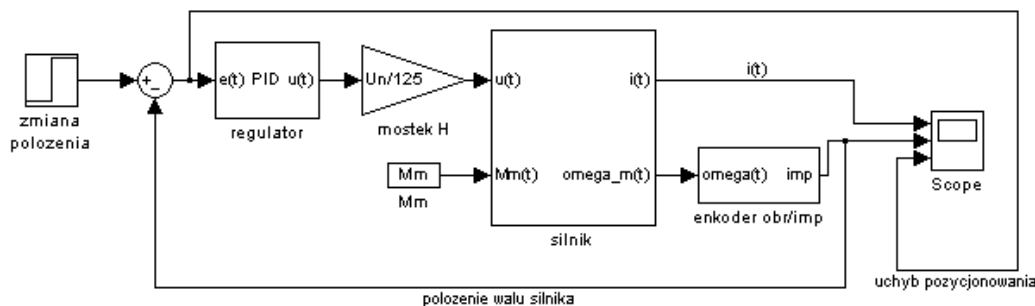
impulsowego, a uzyskana w ten sposób rzeczywista rozdzielczość odczytu będzie równa czterokrotności rozdzielczości enkodera.



Rys. 6. Rozdzielczości przetwornika dla różnych mnożników odczytu

## 5. OCENA JAKOŚCI REGULATORA CYFROWEGO

Jednym z najważniejszych działań związanych z uruchomieniem regulatora jest dokonanie odpowiednich nastaw jego parametrów, tak aby uzyskać jak najlepszą pracę regulatora z obiektem regulacji. W celu przeprowadzenia wstępnego doboru nastaw regulatora sporządzono komputerowy model układu kątownego pozycjonowania silnika. Model ten został wykonany w środowisku Matlab-Simulink (rys. 7).



Rys. 7. Model układu w programie Matlak-Simulink

Dla dokonania oceny jakości regulacji przyjęto następujące kryteria:

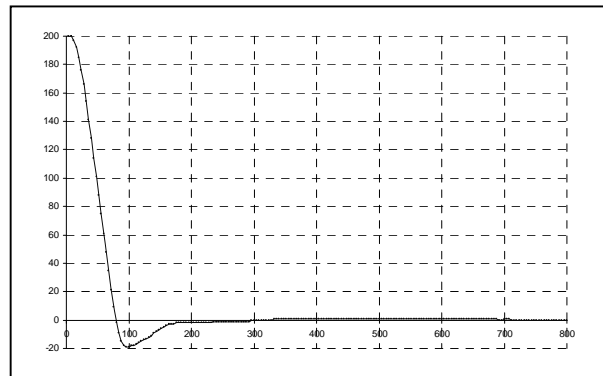
- dla stanów ustalonych – stabilność oraz uchyb statyczny ( $e_{st}$ )
- dla stanów dynamicznych – czas regulacji ( $T_r$ ), odchylenie maksymalne ( $e_{max}$ ).

Wielkością wejściową było żądanie dokonania obrotu wału o określoną liczbę impulsów enkodera. Zadaniem było uzyskanie nastaw przy których regulator zachowa stabilność i zminimalizuje uchyb. Zastosowano trzy metody doboru nastaw regulatora PID: A) Zeiglera-Nicholsa bazującą na znajomości dynamiki obiektu regulowanego, B) odpowiedzi częstotliwościowej wg Zeiglera-Nicholsa, C) specjalnie opracowanej metody będącej modyfikacją zasad przedstawionych w punkcie (A). Wyniki pomiarów wskaźników zawarte zostały w tab. 2, natomiast przykładowy wykres uchybu na rys. 8.

**Tab. 2.** Wskaźniki jakości regulacji

Typ Reg.	M-model R-ukł. Rzeczywisty	stabilny [tak/nie]	$e_{st}$ [imp]	$e_{max}$ [imp]	Tr lub $T_{e_{st}}$ [s]	IAE	czas pomiaru [s]
<i>metoda Zeiglera-Nicholsa bazująca na znajomości dynamiki obiektu regulowanego</i>							
PID	M	T	-1	-80	>0,24	>13,6	2
	R	T	-6	-103	>0,34	>26,2	2
<i>odpowiedzi częstotliwościowej wg Zeiglera-Nicholsa</i>							
PID	M	T	-1	-87	>0,3	>13,8	2
	R	T	0	-96	0,436	15,5	2
<i>metoda dedykowana</i>							
PID	M	T	0	-18	0,68	13	0,8
	R	T	0	-19	0,7	10,8	0,8

Analiza danych tabelarycznych i przedstawionego na wykresie uchybu określa układ regulacji jako stabilny, a przebieg uchybu ma charakter aperiodyczny. Dla zastosowanej dedykowanej metody doboru nastaw regulatora układ charakteryzuje się: • zerowym uchybem  $e_{st}$  w stanie ustalonym • znacznie mniejszą wartością odchylenia maksymalnego uchybu ( $e_{max}$ ) • krótszym czasem regulacji  $Tr$  • mniejszą całką z modułu uchybu  $IAE$  niż w przypadku wyboru metod uniwersalnych.

**Rys. 8.** Przebieg uchybu w układzie

## 6. PODSUMOWANIE

Zaprezentowany w pracy układ mikroprocesorowego regulatora dzięki programowej metodzie tworzenia algorytmu umożliwia dowolną zmianę parametrów konfiguracyjnych regulatora PID (nastaw). Dzięki temu możliwe jest wpływanie na jego zachowanie i pełna akwizycja danych pomiarowych – śledzenie stanu regulatora włącznie z wyłączaniem członów regulatora. Opracowany model matematyczny układu pozwala na przeprowadzanie porównania zachowania się regulatora w środowisku rzeczywistym i wirtualnym.

### Literatura:

- [Brzózka, 04] Brzózka J.: Regulatory i układy automatyki. Mikom 2004.  
 [Welk, 06] Welk A., „Układ komputerowego sterowania kątowym położeniem wirnika silnika prądu stałego, wykorzystujący mikroprocesorowy regulator”, praca dyplomowa magisterska, Wydział Inżynierii Produkcji PW, Warszawa 2006.