

SIEMENS

Ingenuity for life

Sprzężenie krzywkowe

Motion Control

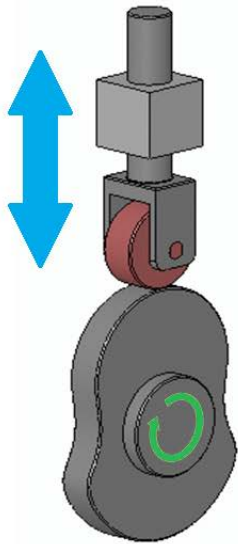
Sprzężenie krzywkowe

siemens.pl/s7-1500T

19 Mechanizm krzywkowy

Znany już w XV wieku mechanizm krzywek jest układem mechanicznym pozwalającym na zamianę ruchu obrotowego (*oś master*) na nieliniowy ruch posuwisto-zwrotny (*oś slave*).

Krzywka jest elementem mimośrodowym konstrukcji mechanicznej o specjalistycznym kształcie przenoszącym ruch wału na tzw. popychacz.



Krzywka mechaniczna

Przez wiele dekad mechanizm krzywkowy był stricte układem mechanicznym dającym szerokie możliwości przekazania pozycji jednego elementu na inny. Zmiana położenia może być jednorazowa lub okresowa. Przeznaczenie takich rozwiązań jest bardzo szerokie, sztandaryowym przykładem zastosowania układu krzywkowego jest mechanizm rozrządu w silnikach spalinowych, stosowany do synchronicznego sterowania pracą zaworów.

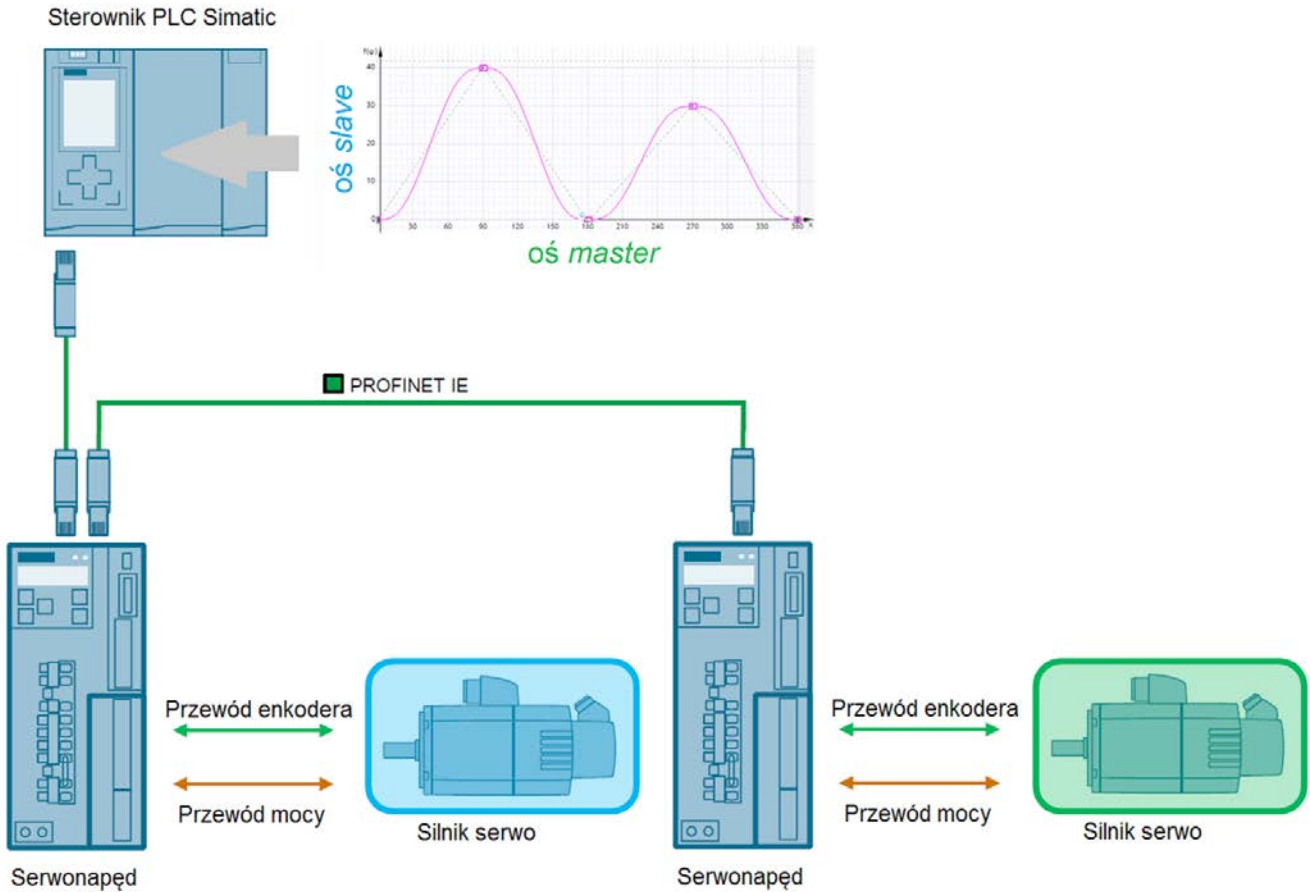
Z punktu widzenia projektu układu krzywkowego – zdefiniowanie wzajemnych przemieszczeń poszczególnych komponentów urządzenia to etap wstępny. Po wykonaniu konstrukcji mechanicznej należy wziąć pod uwagę obciążenie oraz eksploatację elementów wykonawczych, dopuszczalne przyspieszenia i potencjalne źródła błędów. Inżynier musi więc obliczyć precyzyjnie ścieżkę ruchu dla każdego z podzespołów aby zapewnić ciągłość pracy oraz zminimalizować zużycie mechanizmów. W wielu przypadkach poprawność wykonania projektu, doboru parametrów systemu czy optymalizacji układu napędowego może zostać stwierdzona dopiero po wykonaniu szeregu testów rozruchowych (np. o różnej częstotliwości), a także wytrzymałościowych na przestrzeni czasu.

Jeszcze do niedawna, sprzężenia krzywkowe były intensywnie rozwijaną gałęzią inżynierii mechanicznej. W ostatnich latach tendencja ta uległa jednak diametralnej zmianie. Stało się tak przez bardzo dynamiczny rozwój technologii krzywek elektronicznych.

Krzywka elektroniczna

Wykorzystanie nadrzędnego układu napędowego oraz mechaniczne przeniesienie ruchu jednostki głównej na krzywkowy aktuator wtórny, pozwala na skoordynowany ruch synchroniczny o określonym profilu (funkcji). Rozwiązanie ściśle mechaniczne ma jednak sporo wad – dostrzegalnych zarówno na etapie konstrukcyjnym jak i przy pracy oraz konserwacji maszyny. Jednorazowe zastosowanie projektu mechanicznego, zużycie elementów przeniesienia napędu czy kosztowne naprawy oraz rekonfiguracja – to tylko niektóre z utrudnień.

W związku z powyższymi niedogodnościami - sprzężenie mechaniczne zostało zastąpione krzywką elektroniczną. Zamiast pojedynczego mechanizmu napędowego, zastosowano dwa (lub więcej) niezależne napędy, które zostały zsynchronizowane odpowiednim profilem przez funkcję programową nadrzędnego systemu sterowania. Inaczej mówiąc – użyty został nadrzędny sterownik oraz napędy serwo pracujące w układzie synchronicznym. Jest to dokładnie taki sam sposób sprzężenia elektronicznego osi jak poznaliśmy już w poprzednim rozdziale, z tym, że funkcja sprzęgająca nie jest relacją liniową, a dowolną funkcją zależności osi *slave* (nadażnej) względem osi *master* (wiodącej).



Zastąpienie sprzężeń mechanicznych krzywkami elektronicznymi, znacznie poprawiło elastyczność rozwiązań, zwiększyło wydajność pracy maszyn oraz dało możliwość wprowadzania zmian do projektu na dowolnym jego etapie – bez konieczności przebudowy urządzenia. Skomplikowane mechanizmy mogły zostać zastąpione jednym lub wieloma silnikami, które wykonywały odpowiedni ruch skoordynowany według zdefiniowanej funkcji. Kluczowa z punktu widzenia operatora okazała się możliwość zmiany

produktu lub jego wariantu bez zatrzymania, a nawet spowolnienia produkcji dzięki możliwości swobodnej parametryzacji krzywek w trybie pracy systemu.

Cechy te dają nieograniczone możliwości projektowe, prowadzą również bezpośrednio do redukcji kosztów oraz znacząco ułatwiają programowanie oraz usprawniają serwisowanie maszyny.

20 Krzywki w SIMATIC S7-1500T

Zgodnie z opisem z rozdziału poprzedniego – sterownik technologiczny SIMATIC S7-1500T pozwala na programową realizację trzech zasadniczych typów relacji synchronicznych: opisanych już wcześniej sprzężeń liniowych (relatywnego oraz absolutnego), a także krzywkowego. W zakresie rodziny sterowań S7-1500 – tylko jednostka technologiczna daje możliwość realizacji tego ostatniego zadania. Poza koniecznością posiadania jednostki z oznaczeniem „T” nie ma tutaj żadnych innych

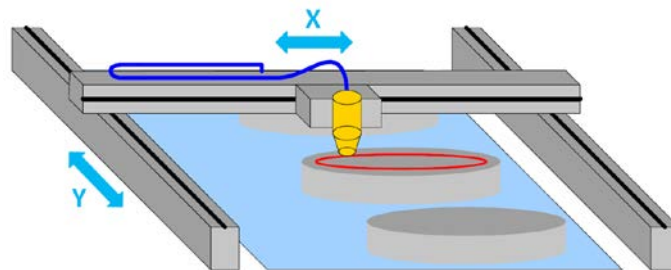
wymagań. Wystarczy środowisko inżynierskie w wersji V14 lub nowszej.

Nie są wymagane żadne dodatkowe pakiety programowe ani komponenty sprzętowe – komunikacja napęd-sterownik realizowana jest przez zintegrowany port PROFINET z obsługą komunikacji w trybie izochronicznym. Część logiczna również „zaszyta” jest w jednostce centralnej.

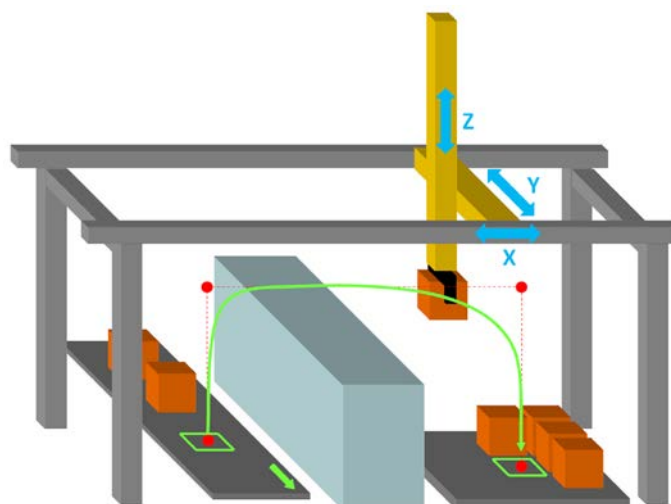
20.1 Zastosowanie krzywki elektronicznej

Mechanizm krzywkowy ma bardzo szerokie zastosowanie w różnego rodzaju aplikacjach. Wszędzie gdzie wymagana jest nieliniowa relacja synchroniczna dwóch lub większej ilości osi możemy zastosować sprzężenie krzywkowe. Przykładem może być wspomniana już w poprzednich dokumentach aplikacja piły latającej lub noża obrotowego, które zrealizować można zarówno przez prostą synchronizację

osi, ale również przez stworzenie odpowiedniej funkcji dla sprzężenia krzywkowego. Charakterystycznym przykładem mogą być również prasy lub urządzenia pakujące typu *flowpack*. Bardziej zaawansowane aplikacje, które również można zrealizować przez odpowiednią konfigurację relacji krzywkowych to urządzenia do kreślenia lub wycinania kształtów (np. frezarka czy wycinarka laserowa lub wodna) w układzie kartezjańskim XY

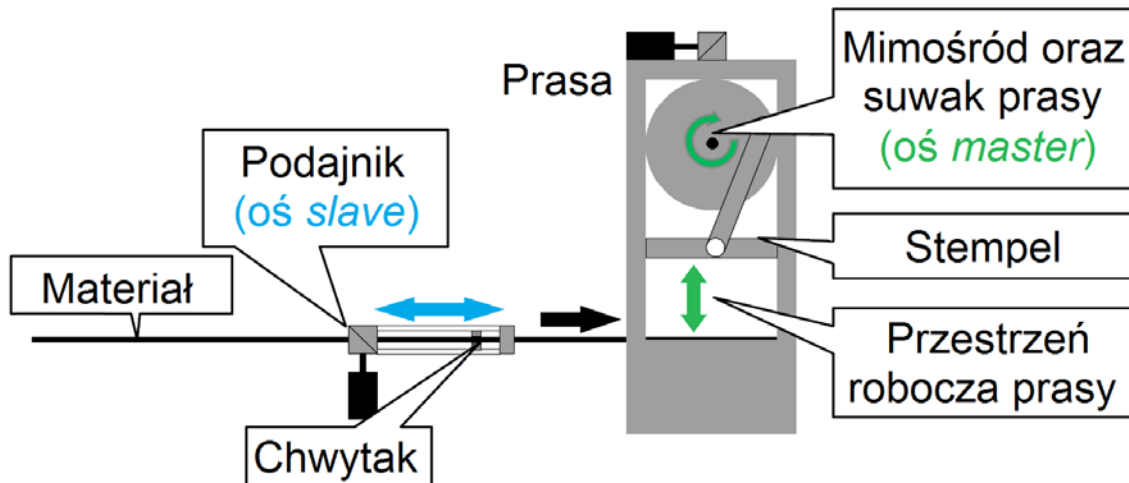


lub prosta interpolacja ścieżki w układzie kartezjańskim XYZ.



Wróćmy do przykładu z początku dokumentu, gdzie ruch obrotowy zostaje przemieniony w posuwisto-zwrotny. Klasyczne zastosowanie takiej konwersji można zaobserwować w prasach. Aplikacja prasy mimośrodowej (rysunek) to doskonała ilustracja takiego mechanizmu. Ruch obrotowy mimośrodowo przekłada się na przesunięcie stempla prasy, który dociskany jest do formy w momencie,

gdy podajnik podsunie materiał w przestrzeń roboczą prasy. Mimośród stanowi tutaj klasyczne, mechaniczne przeniesienie ruchu obrotowego na liniowy, natomiast napęd mimośrodowo oraz napęd podajnika to dwie niezależne jednostki, które synchronizowane są przez profil krzywkowy zdefiniowany w nadrzędnym sterowniku PLC.

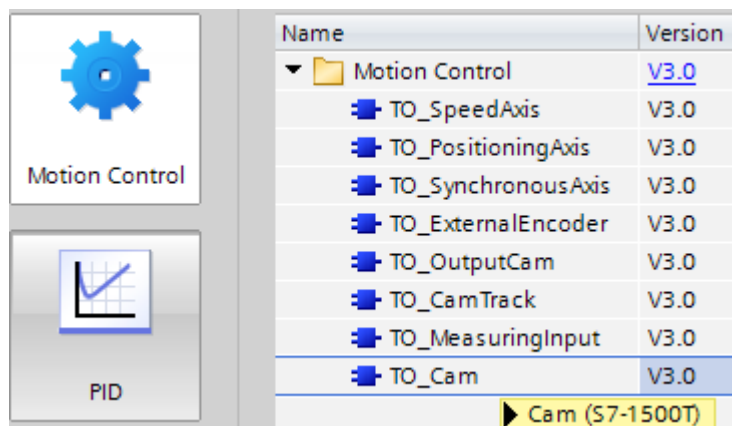


W takim urządzeniu, na jeden obrót napędu prasy (oś *master*) należy wygenerować przesunięcie podajnika od pozycji wyjściowej do pozycji końcowej (w przestrzeni roboczej prasy), a następnie jego powrót do pozycji bazowej. Cała maszyna pracuje zgodnie z cyklem pracy napędu *master*. W dalszej części dokumentu dowiemy się jak skonfigurować krzywkę do takiej aplikacji.

20.2 Obiekt technologiczny *TO_Cam*

Podobnie jak w poprzednich przypadkach, tutaj również – pierwszym krokiem jest dodanie do projektu obiektu technologicznego. Nowy obiekt, który musimy wykorzystać to profil CAM, który de facto służy do sprzężenia dwóch uprzednio utworzonych obiektów technologicznych –

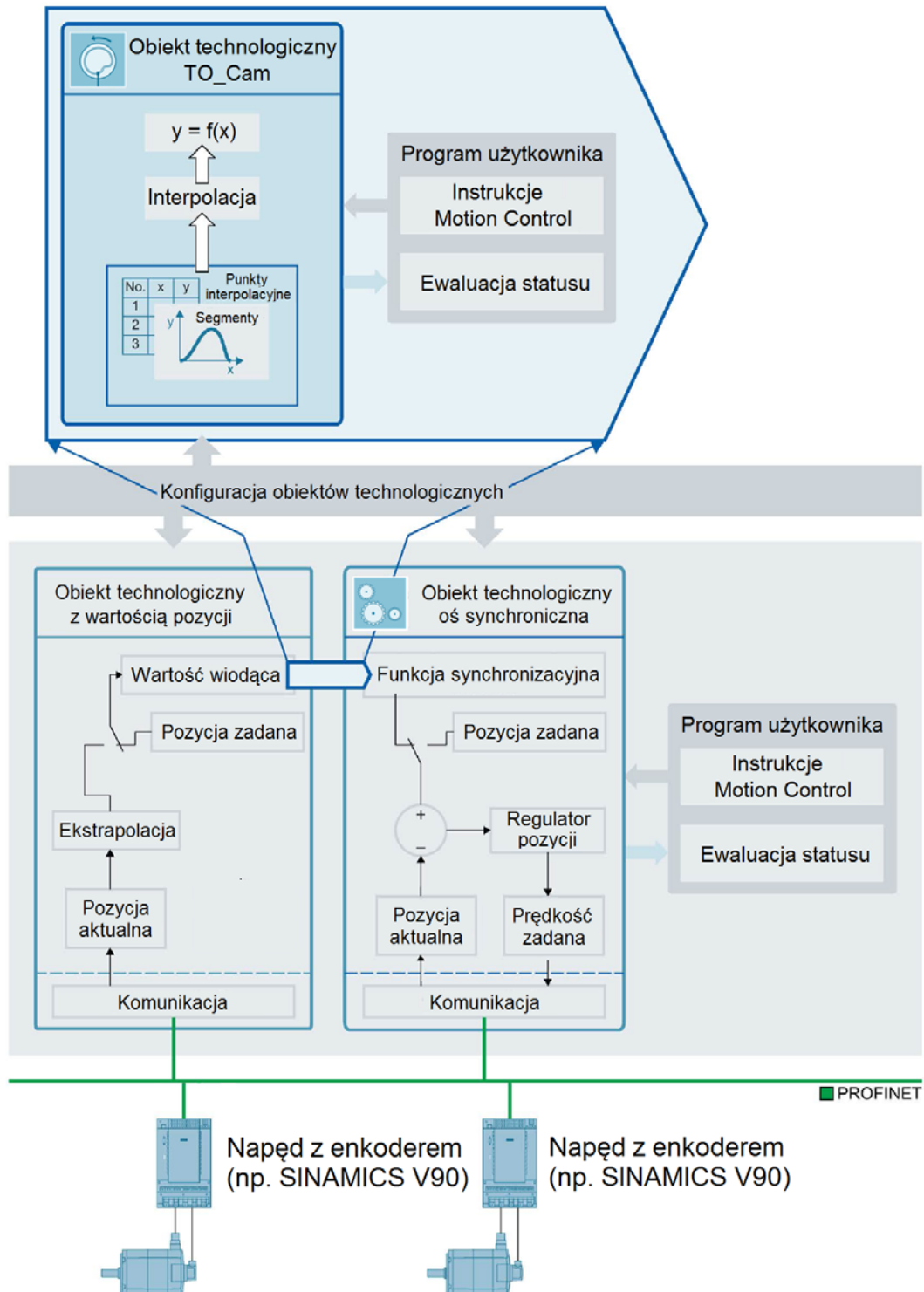
osi *master* oraz osi *slave*. Oś wiodąca może być tutaj osią pozycjonującą, synchroniczną lub enkoderną zewnętrzną. Oś nadążna to – podobnie jak w przypadku każdego innego sprzężenia – oś synchroniczna.



Obiekt technologiczny CAM (krzywka) definiuje funkcję transferową $f(x)$. Zależność wartości wyjściowej względem wejściowej opisana jest przez funkcję o neutralnej jednostce. Funkcję przejścia (profil krzywki) definiujemy przez punkty i/lub segmenty interpolacyjne w dedykowanym edytorze graficznym. Zdefiniowana zależność musi zostać interpolowana w programie użytkownika przez funkcje Motion Control. Ustawienia mogą zostać zmodyfikowane w trybie pracy systemu. Te oraz inne kroki konfiguracyjne zostaną szczegółowo opisane w dalszej części dokumentu.

Poniższy schemat przedstawia działanie mechanizmu synchronizacji osi (obiektów technologicznych) w sprzężeniu krzywkowym realizowanym przez nadrzędny sterownik S7-1500T.

Możemy więc powiedzieć, że w pierwszej kolejności należy spełnić wszystkie wymagania dla pracy synchronicznej dwóch osi, a następnie definiujemy nieliniową relację pozycji pomiędzy tymi obiektami technologicznymi.

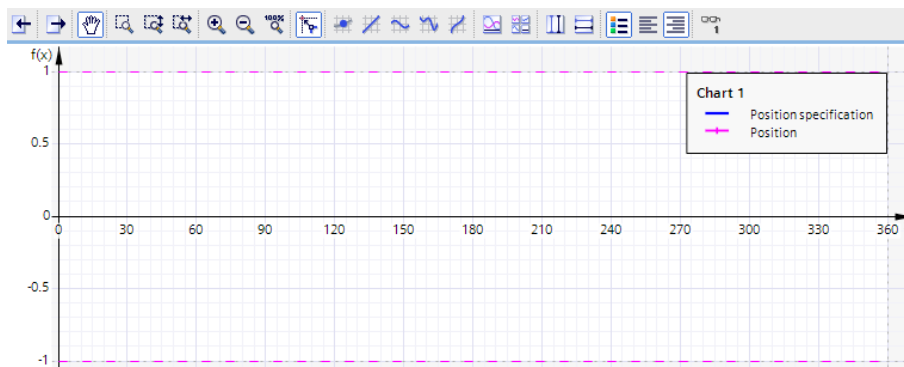


20.3 Edytor krzywek

Edytor zależności pozycji osi nadążnej względem osi wiodącej, pozwala na graficzną oraz tabelaryczną konfigurację punktów kluczowych, przez które ma przebiegać wykres funkcji, a także daje możliwość zaawansowanego określenia interpolacji jaka pomiędzy tymi punktami (lub segmentami) ma zostać przez system zastosowana. Użytkownik ma również możliwość wskazania konkretnej funkcji łączącej dane punkty profilu.

Dzięki elastycznej obsłudze interfejsu użytkownika, swobodzie wprowadzania charakterystyki wejściowej, natychmiastowej analizie parametrów dynamicznych układu w środowisku inżynierskim – zyskujemy wyższą jakość ruchu, a także pełną elastyczność w konfiguracji naszego systemu.


Po otwarciu konfiguratora widzimy pustą przestrzeń roboczą jak na poniższym obrazku.




Podstawowymi parametrami, od których należy rozpocząć konfigurację jest zakres wartości jakie w obszarze sprzężenia krzywkowego przyjmować będzie oś wiodąca oraz nadążna. Nasza przykładowa aplikacja składa się z wiodącej osi obrotowej modulo (0 - 360°) oraz osi liniowej. Takie też ustawienia możemy narzucić dla

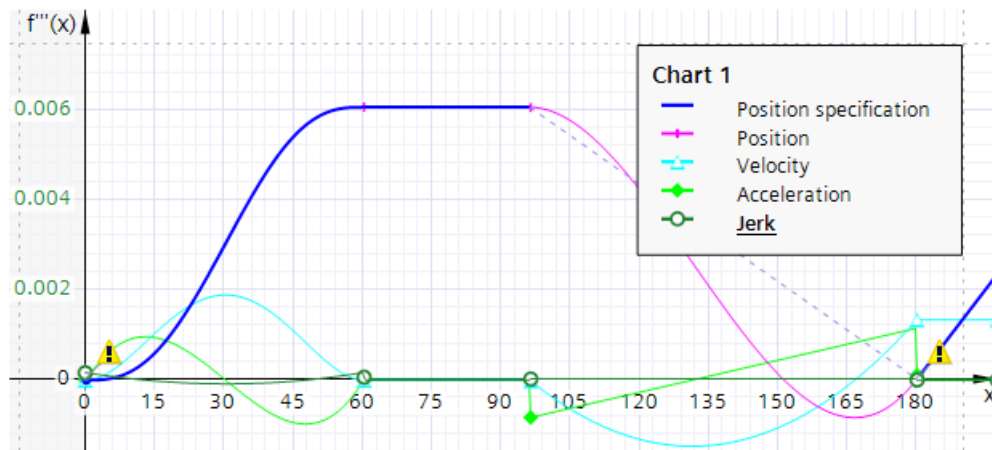
obszaru rysowania (zakładka *Profile* → *General*). Wartości pozycji interpretowane są zgodnie z konfiguracją obiektów technologicznych. Należy upewnić się, że zdefiniowane wartości osi zgadzają się z zakresem ich pracy w układzie mechanicznym aby zdefiniowana funkcja nie spowodowała problemów przy uruchomieniu.

Profil pozycji oraz jego pochodne

W celu zdefiniowania przebiegu funkcji pozycji możemy korzystać z punktu, linii, sinusoidy, wielomianu oraz odwrotnej sinusoidy .

Po wykreśleniu profilu krzywki możemy wybrać opcję wyświetlenia pochodnych funkcji pozycji – na jednym lub czterech osobnych trendach . Z matematyki wiemy, iż pochodną funkcji jest zależność prędkości jej zmian względem jej argumentów. **Argumentem** naszej funkcji pozycji

osi nadążnej jest **pozycja** osi wiodącej. W związku z powyższym **pierwszą pochodną** profilu pozycji $f(x)$ jest funkcja **prędkości** $f'(x)$ (*velocity*), której argumentem będzie niezmiennie pozycja osi wiodącej. **Drugą pochodną** pozycji $f''(x)$ to relacja zmian prędkości względem pozycji czyli **przyspieszenie** (*acceleration*). **Trzecia** natomiast $f'''(x)$ to dynamika przyspieszenia czyli tzw. **zryw** (*jerk*).



System oblicza i przedstawia powyższe przebiegi w formie graficznej w celu optymalizacji pracy układu. Funkcje te wykorzystywane są również celem

interpolacji oraz mogą być wyznacznikiem ciągłości w celu wygładzenia profilu pozycji krzywki.

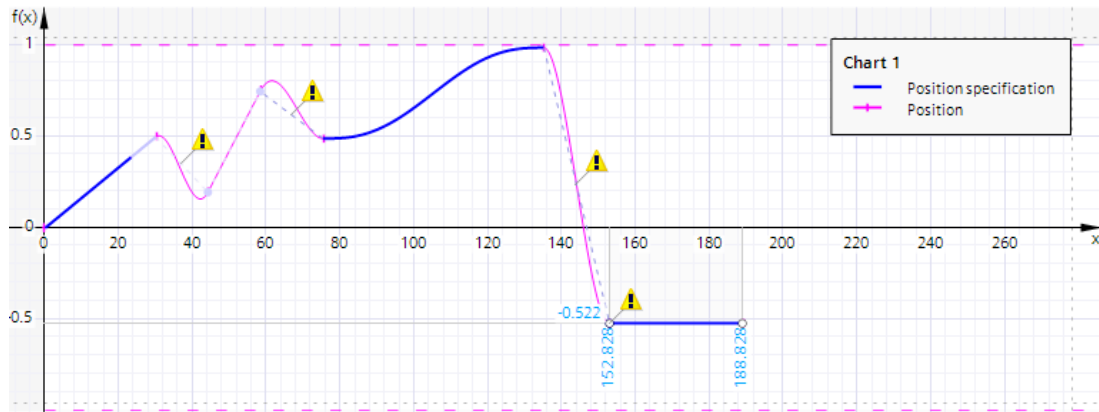
Interpolacja segmentów przejściowych

Rysując przebieg profilu krzywki korzystając z edytora graficznego – możemy wykonać go w sposób ciągły (bez luk wartości w pełnym zdefiniowanym zakresie) lub wsta-

wiając jedynie segmenty (lub punkty), które są kluczowe dla naszej aplikacji. Odcinki nieokreślone mogą finalnie zostać zdefiniowane przez wybór funkcji matematycznej w tabeli edytora.

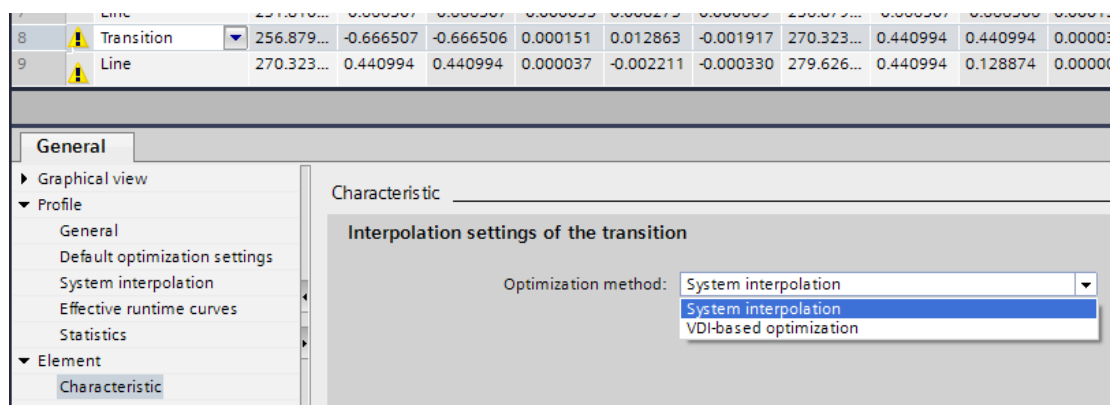
Line	ou
Transition	96
Point	18
Line	18
Sine	18
Transition	25
Inverse sine	28
Polynomial	31

Ustawiając jednak funkcję **segmentów przejściowych** pomiędzy określonymi funkcjami na domyślnej opcji *Transition* - pozostawiamy ich interpolację mechanizmom systemowym.



Kolejne dwie zakładki w konfiguracji ogólnej profilu (*Default optimization settings* oraz *System interpolation*) określają parametry wybranego algorytmu optymalizacji (interpolacji) dla tych właśnie niezdefiniowanych segmentów przejściowych.

Dla każdego z nieokreślonych segmentów użytkownik może wybrać spośród dwóch metod interpolacji – VDI lub systemowej SIMATIC. Wyboru należy dokonać w konfiguracji danego segmentu czyli po jego zaznaczeniu w ostatniej zakładce edytora *Element*.



Jeżeli system w trakcie analizy tworzonego profilu ma jakieś ostrzeżenia dla poszczególnych elementów funkcji – informuje o tym bezpośrednio na wykresie przez wyświetlenie ikony ostrzegawczej ⚠. Skierowanie kursora myszy na symbol wykrzyknika spowoduje wyświetlenie komunikatu systemowego. Mogą być to informacje dotyczące nieciągłości wykresu (lub

nakładania się punktów), problemów z kalkulacją efektywnych pochodnych, przekroczeń zakresów lub inne informacje dotyczące diagnostyki profilu. Nie ma konieczności usuwania wszystkich ostrzeżeń aczkolwiek jeśli jest to możliwe to warto to wykonać – z pewnością poprawi to jakość tworzonej funkcji.

Interpolacja systemowa

Wybierając opcję *System interpolation* jako metodę optymalizacji segmentu, należy określić algorytm jaki ma zostać zastosowany. Wyboru algorytmu dokonujemy w zakładce ustawień ogólnych profilu (*Profile* → *System interpolation*). Optymalizacja systemowa SIMATIC pozwala na interpolację liniową, funkcjami sklejanymi stopnia trzeciego oraz krzywymi Béziera.

Korzystając z interpolacji systemowej możemy określić również charakterystykę segmentów skrajnych (*Behavior at boundary*) istotną przy rekurencyjnym wywołaniu profilu. Krzywka zostaje interpolowana w taki sposób aby jej pierwsza pochodna (prędkość) była równa na początku oraz na końcu profilu.

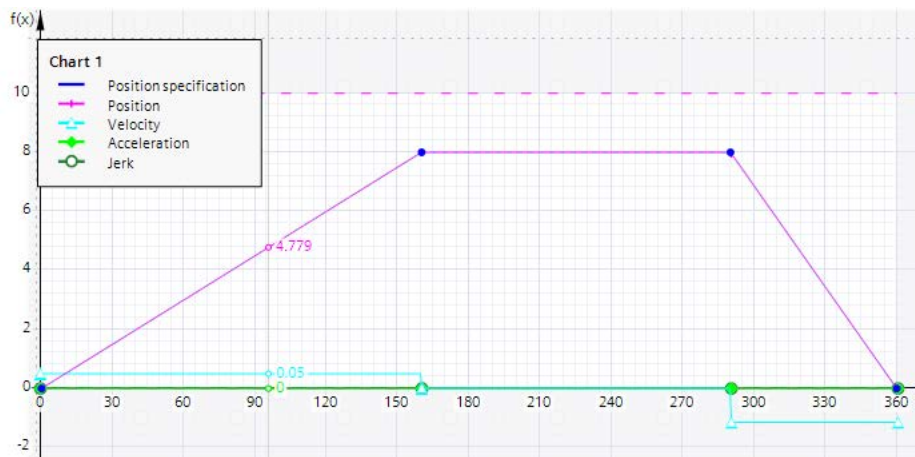
Rozważmy przykładowy profil krzywki, który został zdefiniowany przez 4 punkty w obszarze roboczym edytora graficz-

nego. Wstawione punkty wyznaczają nam 3 segmenty, które wymagają interpolacji. Przykład ten może być bezpośrednim nawiązaniem do rozważanej aplikacji prasy, gdzie w pierwszej fazie cyklu – podajnik z chwytakiem przesuwa materiał (np. blachę) w obszar roboczy prasy, tam – faza druga – prasa wykonuje swoje zadanie (np. wytłoczenie, zgrzewanie lub wycięcie), a w etapie ostatnim – powrót chwytaka do pozycji wyjściowej. Wszystkie etapy pracy osi nadążnej skoordynowane są w trakcie jednego obrotu wiodącej osi rotacyjnej prasy mimośrodowej.

W celu przedstawienia działania algorytmów optymalizacyjnych – wstępnie nie będziemy określać odcinków dostępnymi funkcjami. Uwzględnimy również funkcję wygładzania pierwszej pochodnej na odcinkach skrajnych profilu – będzie miało to znaczenie przy optymalizacji nieliniowej (w cyklicznej pracy systemu).

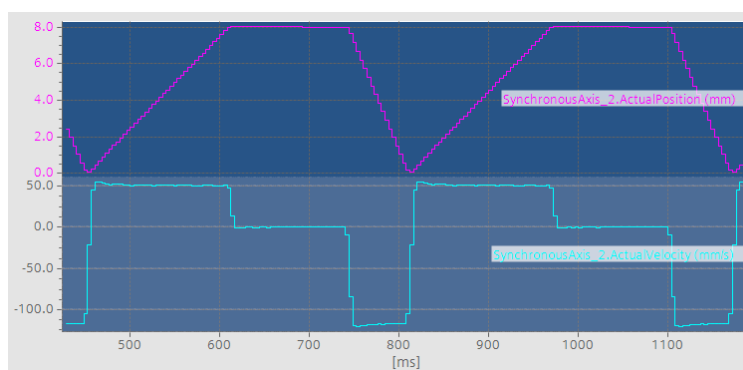
Interpolacja liniowa (*Linear interpolation*)

Jest to szczególny przypadek aproksymacji za pomocą funkcji liniowej. W nawiązaniu do wcześniejszego opisu – w celach testowych – wygenerujemy przebieg profilu pozycji wraz z jego pochodnymi (**pozycja**, **prędkość**, **przyśpieszenie**, **zryw**).



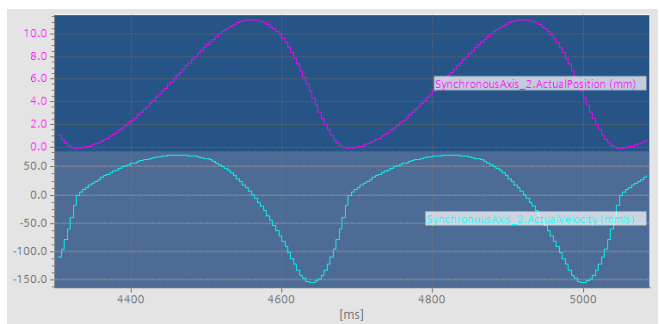
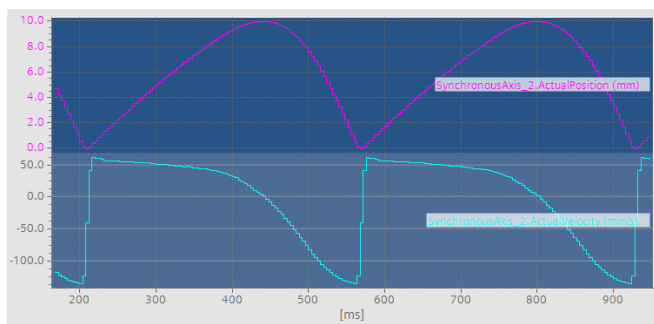
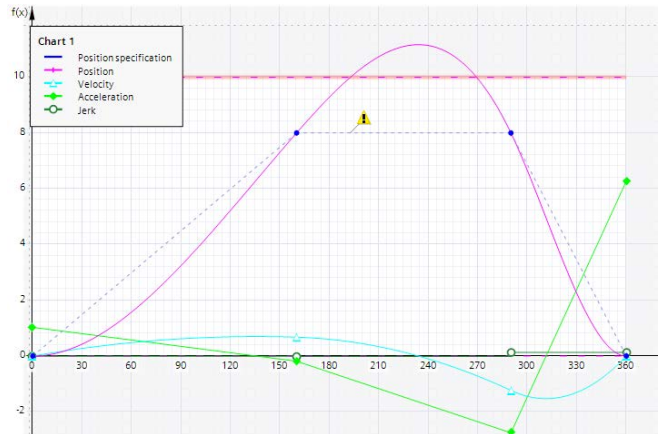
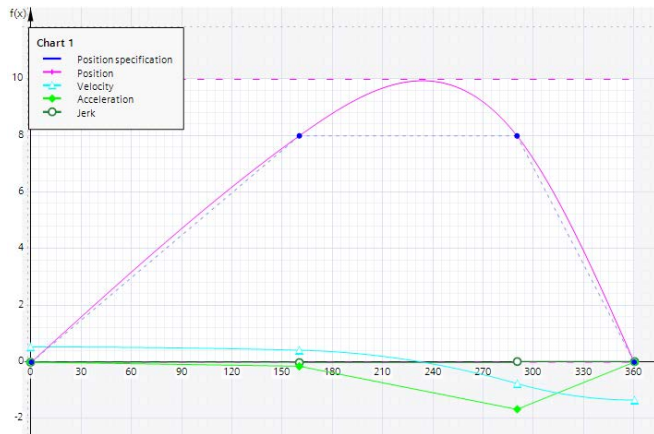
Metoda jest bardzo prosta i nie generuje przekroczeń limitów aczkolwiek w naszym przypadku powoduje nagłe zmiany dynamiki osi nadążnej, co będzie skutkowało szarpnięciem układu (zwłaszcza przy wyższej prędkości). Poniżej przykładowy rzeczywisty przebieg **pozycji**

oraz **prędkości** osi nadążnej w czasie - dla powyższego rozwiązania wywołanego cyklicznie. Możemy w łatwy sposób zaobserwować przeniesienie ustawień profilu na realny system.



Interpolacja funkcjami sklejanymi stopnia trzeciego
(Interpolation with cubic splines)

Jest to metoda numeryczna polegająca na przybliżeniu nieznannej funkcji wielomianami niskiego stopnia. Algorytm ten jest często preferowany względem interpolacji wielomianowej ze względu na to, iż jego zastosowanie minimalizuje błędy aproksymacji.



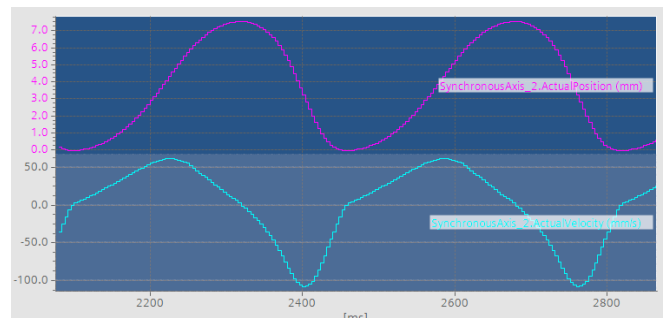
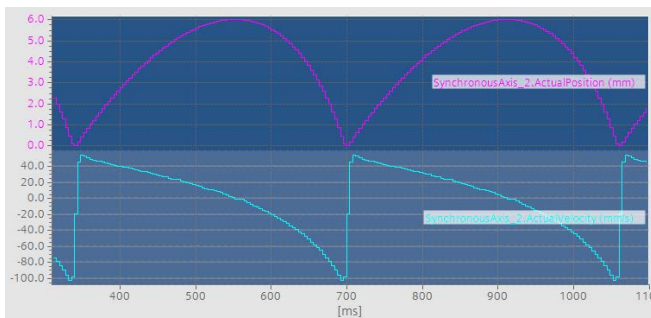
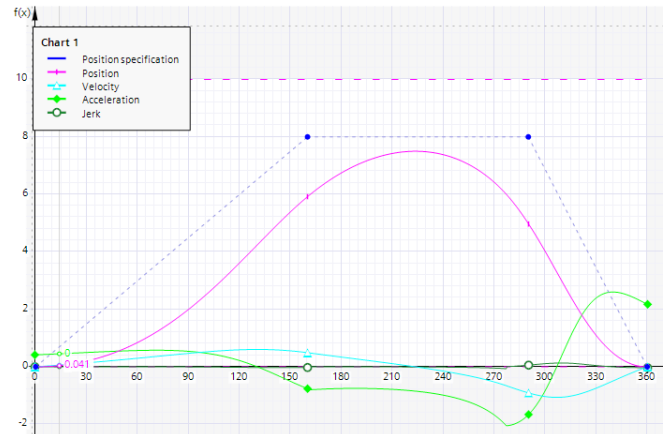
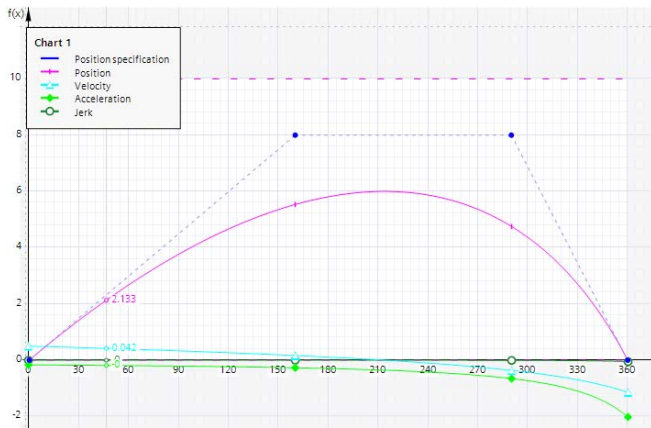
W naszym przykładzie algorytm sprawdzi się o wiele lepiej niż aproksymacja liniowa – zmiana pozycji będzie znacznie bardziej wygładzona, aczkolwiek możemy zaobserwować, iż w przypadku bezwarunkowej interpolacji (strona lewa) - podczas rekurencyjnego wywołania krzywki - nastąpi międzyfazowa nagła zmiana

kierunku pracy osi (szarpnięcie). Z kolei w przypadku uwarunkowania ciągłości **pierwszej pochodnej** (strona prawa) – przejście do kolejnego cyklu jest gładkie, zwróćmy jednak uwagę, iż wyznaczona funkcja przekroczyła wyznaczony przez nas zakres wartości osi nadążnej.

Interpolacja krzywymi Bézier-a (Interpolation with Bézier splines)

Algorytm ten został opracowany w latach 60-tych przez francuskiego inżyniera firmy Renault. Krzywe Bézier-a są krzywymi parametrycznymi o charakterystyce wielomianowej. Znajdują szerokie zastosowanie w systemach optymalizacyjnych przetwarzania grafiki. Ze względu na cechy funkcji opisującej kształty, świetnie nadaje się także przy zagadnieniach optymalizacyjnych układów krzywkowych Motion Control.

Charakterystyczne dla optymalizacji krzywymi Bézier-a jest fakt, iż wyznaczona funkcja nie przebiega przez wszystkie punkty wskazane do interpolacji. W przypadku 4 punktów określonych w naszym przykładzie – algorytm wyznacza krzywą (3-ciego stopnia), która jest ograniczona przez jej styczne (w punkcie 1 oraz 4) wyznaczone przez pary punktów 1-2 oraz 3-4 (lewa strona – bez utrzymania ciągłości pierwszej pochodnej na skrajnych odcinkach profilu).



W naszym przykładzie ta metoda interpolacji wypada zdecydowanie najkorzystniej (zwłaszcza z uwzględnieniem utrzymania ciągłości prędkości na skrajnych odcinkach przebiegu – strona prawa) – nie powoduje przekroczeń zakresów wyznaczonych dla osi nadążnej, a także generuje optymalnie wygładzony kształt profilu. W związku z charakterystyką algorytmu wadą jednak może okazać się fakt, iż punkty 2 oraz 3 nie zostały ujęte w ścieżce osi nadążnej. Można to jednak zniwelować przez zmianę lokalizacji wyznaczonych wstępnie punktów lub przez połączenie różnych metod optymalizacji w konfiguracji przebiegu krzywki – o czym w dalszej części.

Wybierając interpolację systemową – wskazany algorytm zostaje zaordynowany dla wszystkich segmentów profilu, gdzie funkcja przejścia nie została określona. Aby zastosować inną metodę interpolacji dla różnych (nieopisanych funkcją) segmentów – możemy zastosować optymalizację VDI lub wybrać manualnie funkcję dla segmentu.

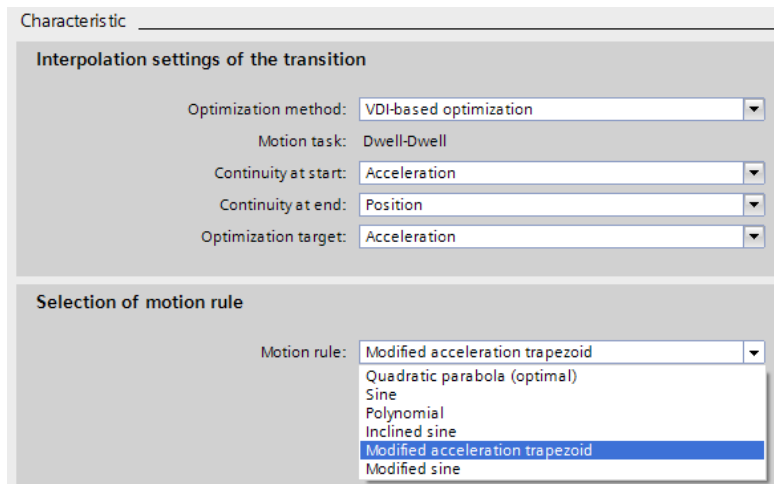
Optymalizacja VDI

Stowarzyszenie Inżynierów Niemieckich VDI (*Verein Deutsche Ingenieure*, www.vdi.eu) zdefiniowało wiele reguł w zakresie automatyzacji przemysłowej. Wśród nich znalazł się dokument (jeszcze z lat 80-tych) dotyczący standaryzacji zasad ruchu w układach ze sprzężeniem krzywkowym (*VDI Guideline 2143 – Motion rules for cam mechanism*). Wytyczne tej normy określają m.in. metodę optymalizacji profilu krzywki. Procedura znalazła miejsce jako alternatywa dla opcji systemowych proponowanych przez firmę Siemens.

Algorytm realizowany jest w oparciu o dwa wyróżniki:

- ciągłość wybranego parametru układu – pozycja, prędkość (bez wstrząsów), przyspieszenie (bez szarpnięć) lub zryw.
- cel optymalizacji (zgodnie z VDI) – brak określonego parametru, prędkość, przyspieszenie, zryw lub minimalna dynamika momentu.

Domyślne parametry optymalizacji VDI możemy skonfigurować globalnie (*Profile* → *Default optimization settings*). Dzięki temu wszystkie segmenty z wybraną interpolacją VDI będą estymowane w ten sam sposób. Z drugiej strony – korzystając z ustawień szczegółowych – każdy segment zdefiniowany dla tej metody interpolacji – sparametryzować możemy niezależnie (*Element* → *Characteristic*). Charakterystyka indywidualnych segmentów, daje również więcej kombinacji określających algorytm interpolujący, np. różnicowanie funkcji ciągłości na początku oraz końcu segmentu czy zastosowanie wybranej zasady określającej ruch (*Motion rule*) będącej częścią specyfikacji *VDI Guideline 2143*. Dostępne opcje filtrowane są w zależności od parametrów wybranych w panelu *Interpolation settings of the transition*.



Więcej informacji odszukać można w tematach pomocy systemowej TIA Portal lub w samej dokumentacji organizacji VDI.

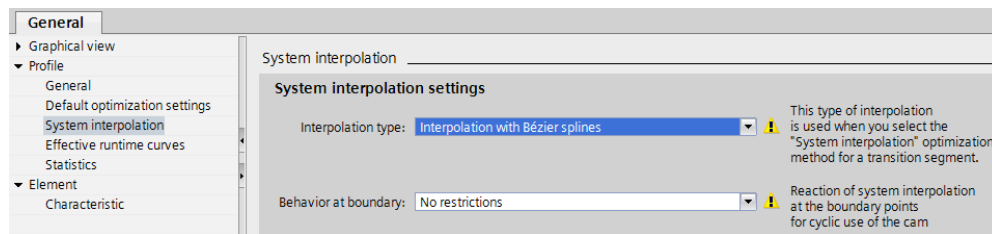
Z praktycznego punktu widzenia – korzystając z algorytmów VDI – uzyskamy efekty bardzo podobne do omówionej w poprzedniej sekcji interpolacji systemowej. W związku z powyższym w niniejszej dokumentacji nie będziemy

zajmować się szczegółową analizą efektów optymalizacji VDI. Warto mieć jednak na względzie, iż pozwala ona (w przeciwieństwie do optymalizacji systemowej) na wybiórczą charakterystykę wskazanych segmentów, co można wykorzystać przy interpolacji mieszanej.

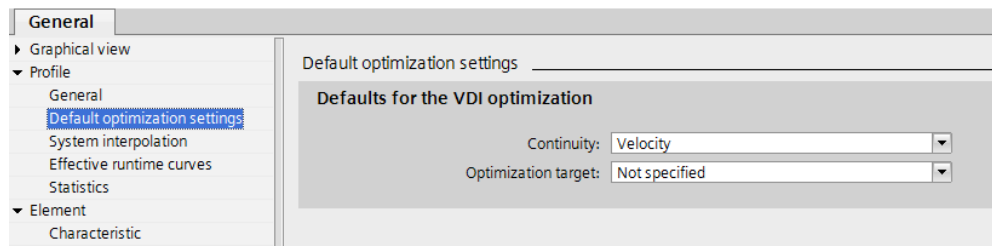
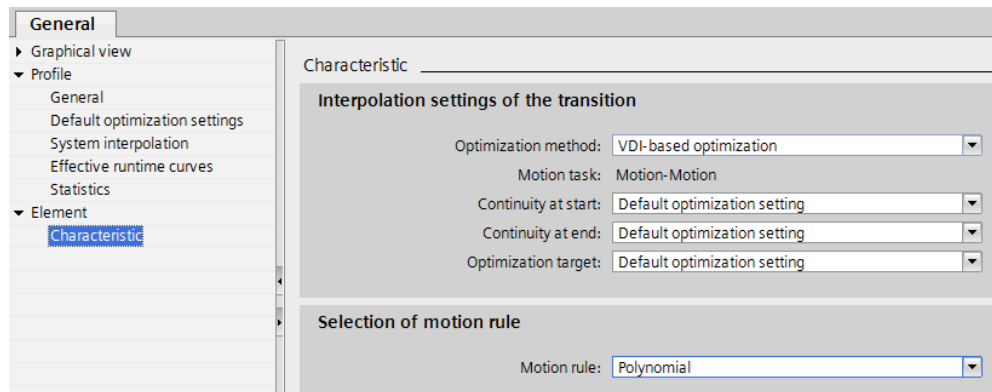
Interpolacja mieszana

Biorąc pod uwagę potrzeby naszej hipotetycznej aplikacji – możemy założyć, iż wskazane będzie chwilowe zatrzymanie obrabianego materiału po jego podsunięciu w obszar roboczy prasy. Założmy więc, że po wykonaniu 160° obrotu osi wiodącej (prasa), nastąpi wykonanie zadania (np. zgrzewanie), co będzie wymagało zatrzymania osi nadążnej (podajnik) na 130° obrotu osi *master*. Po tym etapie następuje powrót podajnika do pozycji wyjściowej (zostaje na tę fazę 70° obrotu osi wiodącej).

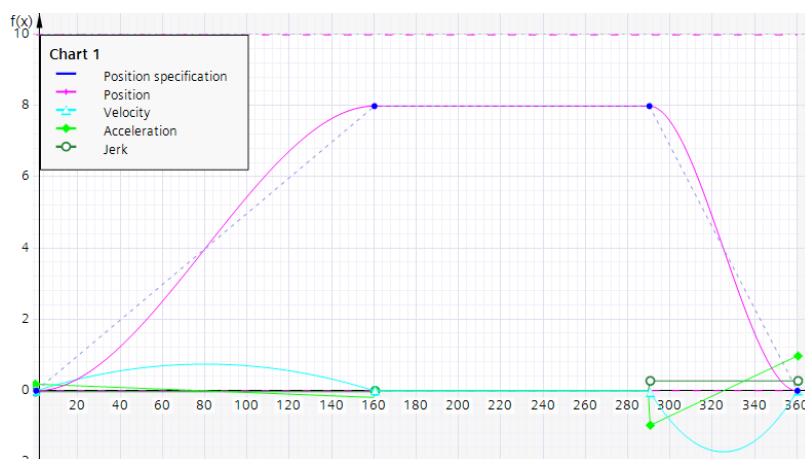
Zastosowanie połączenia dwóch omówionych metod interpolacji pozwoli uzyskać pożądany efekt. Konkretnie – dla segmentu pierwszego (pomiędzy punktami 1-2) oraz dla segmentu trzeciego (wyznaczonego przez punkty 3-4) wybierzmy optymalizację systemową krzywymi Bézier-a (bez wymogu utrzymania ciągłości pierwszej pochodnej na odcinkach skrajnych).



Z kolei dla segmentu środkowego (pomiędzy punktami 2-3) wybierzmy metodę VDI ze wskazaniem utrzymania ciągłości prędkości oraz bez określenia celu optymalizacji.

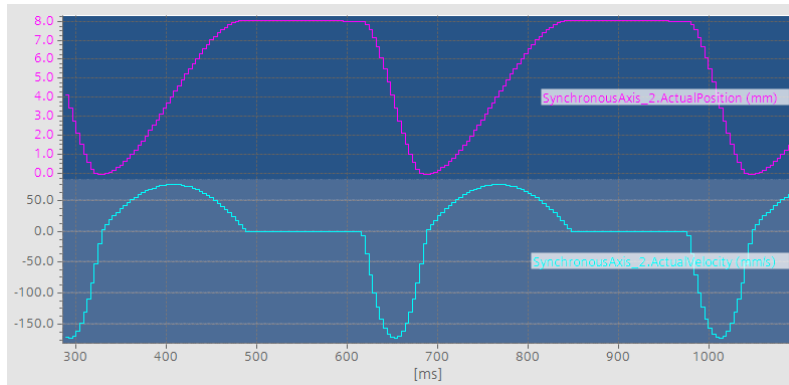


Powyższe ustawienia w zaaplikowane dla naszego 4-punktowego przykładu spowodowały wygenerowanie następującego profilu krzywkowego.



Wynik interpolacji spełnia oczekiwania projektowe i wydaje się być optymalny z punktu widzenia dynamiki układu – nie ma przekroczeń zakresów, profil przechodzi przez wszystkie zdefiniowane punkty, a także prędkość ma charakterystykę

ciągłą. Poniżej – podobnie jak w poprzednich przykładach – zaprezentowany został przebieg **pozycji** oraz **prędkości** osi nadążnej wygenerowany na rzeczywistym układzie dwóch serwonapędów.



Czy rzeczywiście taki profil jest optymalny? Zależy to od wielu czynników. Dopiero wykonanie testów wytrzymałościowych (również długoterminowych) kompletnego syste-

mu – pozwoli określić zużycie elementów, zbadać drgania układu, jego reakcję na zmiany częstotliwości oraz dynamiki pracy czy też oszacować jego żywotność.

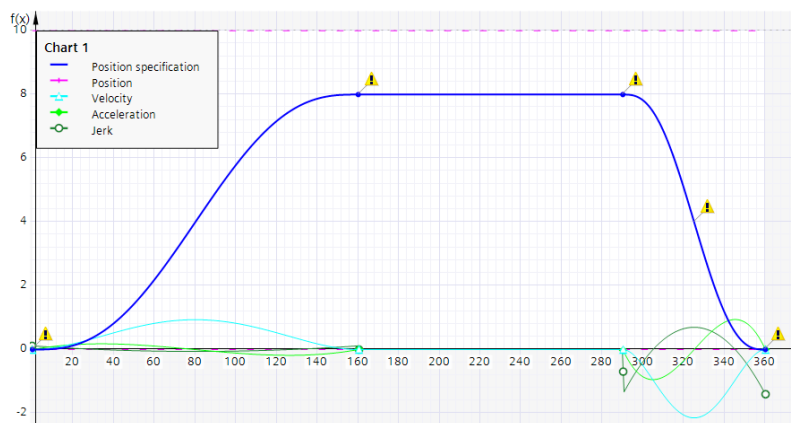
Manualne określenie funkcji segmentów przejściowych

Na początku sekcji wspomnieliśmy o możliwości ręcznego zdefiniowania funkcji matematycznej dla segmentów przejściowych – bez konieczności ich interpolowania. Sprawdźmy więc jaki będzie efekt bezpośredniego wyboru dokonanego przez użytkownika.

stępnych funkcji, możemy określić matematyczną zależność pomiędzy dwoma punktami – dodatkowy punkt, linia, sinusoida bądź wielomian. Jeżeli jedna z tych opcji będzie dla naszego układu odpowiednia, nie będzie konieczności stosowania interpolacji.

Po wstawieniu punktów kluczowych dla naszego profilu – system automatycznie generuje (w tabeli poniżej obszaru rysowania) wiersze dla segmentów przejściowych. Pozostawiając tam wartość domyślną (*Transition*) wykonamy interpolację. Wskazując jednak bezpośrednio jedną z do-

W naszej aplikacji możemy spróbować wybrać wielomian dla obu segmentów skrajnych, natomiast dla środkowego linię. Uwaga – wybór dokonany w tabeli edytora można wycofać jedynie przez usunięcie segmentu z obszaru rysowania lub przez systemową funkcję *Undo*.



Łatwo zauważyć, iż otrzymany profil pozycji jest bardzo podobny do rezultatu interpolacji systemowej połączonej z VDI. Różnice dostrzegamy dopiero przy funkcjach przyspie-

żenia oraz zrywu. Ich istotność określają już testy uruchomieniowe.

Emulator parametrów dynamicznych

W zakładce *Profile* → *Effective runtime curves* możemy skonfigurować parametry krzywej trybu runtime dla osi wiodącej oraz nadążnej. Emulator programu sterownika analizuje przebieg sprzężenia pozycji (oraz jego pochodnych) zgodnie z zadanymi parametrami (skalowanie, jednostki, dynamika) i prezentuje rezultat w obszarze roboczym edytora graficznego.

Statystyka

Warto również zwrócić uwagę na liczbę wykorzystanych elementów konfiguracyjnych, która wskazana jest w ostatniej zakładce właściwości krzywki (*Profile* → *Statistics*). Maksymalnie możemy zdefiniować 1000 punktów oraz 50 segmentów graficznie opisujących profil. Będzie to istotne

Wprowadzone modyfikacje nie są wgrywane do sterownika. Oznacza to, że interpolacja krzywki w CPU przebiega zgodnie z parametrami bazowymi. Wyznaczone współczynniki pozwalają jednak na testowanie oraz wizualizację dynamiki układu, np. symulacja reakcji na skalowanie w programie użytkownika przez parametry funkcji *MC_CamIn*.

w przypadku gdy będziemy chcieli zastosować mechanizmy do modyfikacji profilu w trybie pracy, o czym w dalszej części dokumentu.

20.4 Interfejs użytkownika

Po zdefiniowaniu profilu krzywkowego naszego układu przechodzimy do programu użytkownika, gdzie sprzężenie zostanie uruchomione.

W poprzednim rozdziale (*Ruch skoordynowany*) odnajdziemy wiele analogii w konfiguracji obiektów technologicznych oraz funkcji interfejsu programistycznego użytkownika.

Przed wszystkim należy pamiętać, iż podstawowymi parametrami wejściowymi do naszej relacji krzywkowej są dwa standardowe obiekty technologiczne. Przynajmniej jeden z nich jest osią synchroniczną (oś nadążna), drugi natomiast

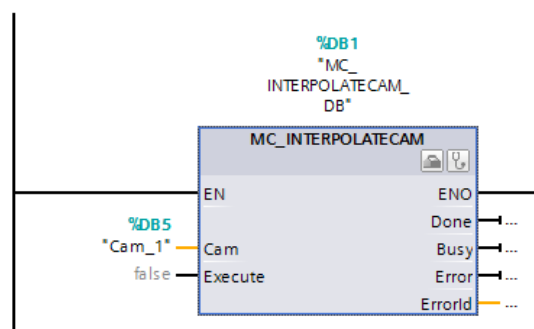
może być osią pozycjonującą, synchroniczną lub enkoderm zewnętrznym. Konfiguracja obiektów technologicznych osi odbywa się identycznie jak w przypadku wcześniej omawianej synchronizacji liniowej – profil krzywkowy jest analogicznym mechanizmem sprzęgającym, z tym, że pozwala na definicję nieliniowej relacji pomiędzy osiami.

Różnica programistyczna jest również znikoma – konfiguracja programu w sterowniku sprowadza się do wywołania niemalże identycznej funkcji sprzęgającej oraz do uprzedniej interpolacji profilu krzywki w programie użytkownika.

Funkcja *MC_InterpolateCam*

Zgodnie z analizą przeprowadzoną w poprzedniej sekcji – efektem pracy graficznego interfejsu środowiska inżynierskiego jest opis trajektorii sprzężenia krzywkowego. Wyznaczyliśmy metodę interpolacji profilu (lub jego statyczny opis funkcjami matematycznymi) – teraz musimy wykonać praktyczną interpolację w programie użytkownika. Wynika

to z faktu, iż wgranie określonego graficznie profilu do sterownika S7-1500T, powoduje jego ulokowanie w tymczasowej pamięci, do momentu jego interpolacji przez funkcję użytkownika. Dopiero wywołanie funkcji *MC_InterpolateCam* spowoduje aktualizację profilu zgodnie z konfiguracją obiektu technologicznego *TO_Cam*.



Podczas interpolacji opisane zostają wszystkie luki pomiędzy punktami oraz segmentami zdefiniowanymi w konfiguracji. Procedura przypisuje stałą wartość pozycji osi nadążnej dla każdej wartości zdefiniowanego zakresu pracy osi wiodącej.

Teoretycznie – wgranie profilu, który w środowisku inżynierskim został opisany kompletnie, czyli zdefiniowane zostały wszystkie wartości osi nadążnej w pełnym zakresie pracy osi wiodącej (bez interpolacji) – nie wymaga zastosowania funkcji *MC_InterpolateCam*, a co za tym idzie spowoduje automatyczną reinicjalizację profilu w sterowniku zaraz po wgraniu nowej wersji projektu. W praktyce jednak, pominięcie chociażby jednego punktu, wymusi wywołanie

funkcji interpolacyjnej. W związku z powyższym dobrą praktyką jest założenie, iż zawsze przed wywołaniem sprzężenia krzywkowego - powinna zostać wykonana interpolacja programowa. Nie jest to procedura, która w jakikolwiek sposób zaburzy pracę systemu lub utrudni konfigurację.

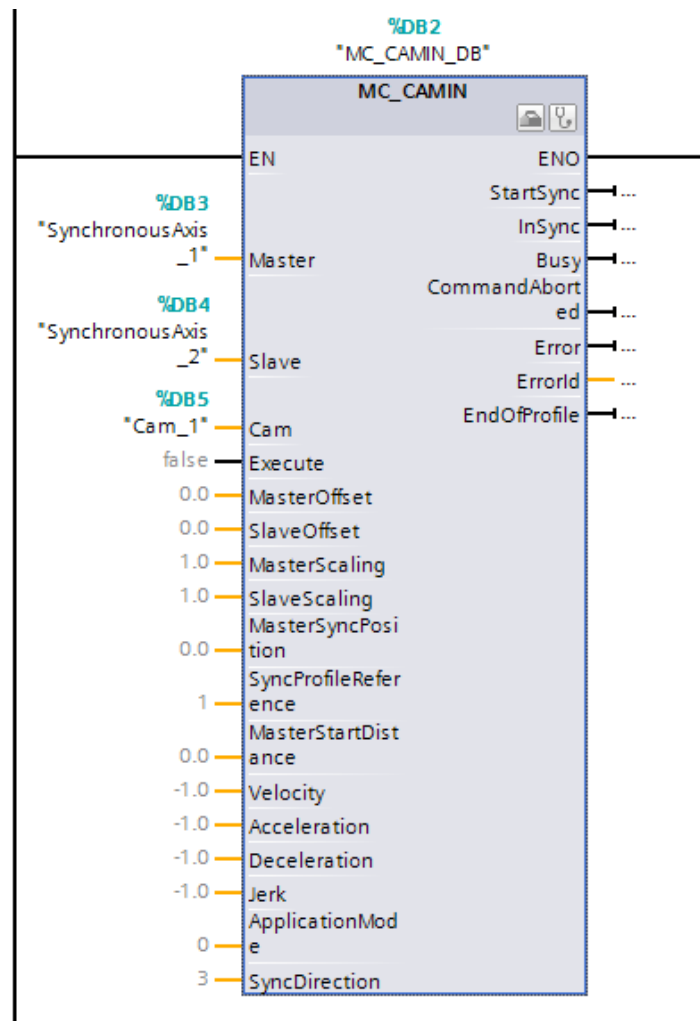
Pod parametr *Cam* podłączamy nasz obiekt technologiczny (można przeciągnąć metodą *drag&drop* z drzewka projektu), oraz podajemy zbcze narastające na wejście *Execute*.

Wyjście *MC_InterpolateCam.Done* dostarczy nam informacji o zakończeniu procedury. Dobrze jest zatem aby sygnał ten wykorzystać bezpośrednio jako wejście *MC_CamIn.Execute* aktywujące sprzężenie krzywkowe.

Funkcja *MC_CamIn*

W poprzednim rozdziale omówiliśmy szczegółowo zasady działania funkcji *GearInPos* służącej do aktywacji liniowej synchronizacji absolutnej osi. Analogicznym sprzężeniem

(z tym, że nieliniowym) jest relacja krzywkowa. W związku z powyższym, funkcja realizująca dojsie do zasprężlenia oraz jego utrzymanie w trybie pracy, ma sporo podobieństw.



Parametry podstawowe

Bazowymi wejściami bloku są w tym przypadku oczywiście nasze osie (*Master*, *Slave*), a także nieliniowa relacja pomiędzy nimi czyli nasz profil krzywkowy (*Cam*). Standardowo wywołanie funkcji odbywa się przez podanie zbocza narastającego na wejście *Execute*.

Skalowanie

Kolejne cztery parametry funkcji *MC_CamIn* (*MasterOffset*, *SlaveOffset*, *MasterScaling* oraz *SlaveScaling*) stanowią swego rodzaju odniesienie do przekładni, którą definiowaliśmy w przypadku liniowego sprzężenia osi. Pozwolą zatem zmodyfikować zakres operacji naszego profilu lub wykonać jego skalowanie. Więcej informacji na temat modyfikacji krzywek w trybie pracy aplikacji przedstawimy w dalszej części dokumentu.

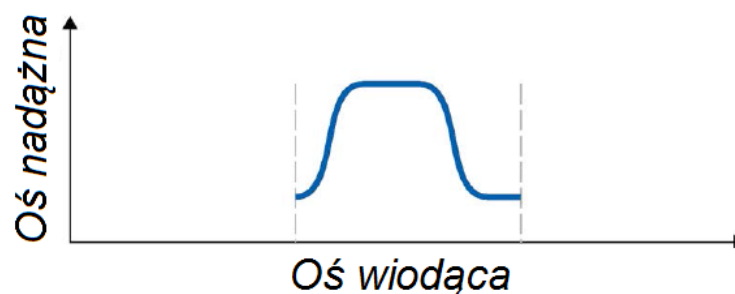
Dojście do zasprzężenia

Podobnie jak w przypadku funkcji *MC_GearInPos* – funkcja *MC_CamIn* pozwala określić parametry dojścia do zasprzężenia. Analogicznie możemy wybrać tutaj synchronizację „na drodze” (*SyncProfileReference* = 1) lub przez zdefiniowanie maksymalnych wartości parametrów dynamicznych (*SyncProfileReference* = 0). Opis tych dwóch trybów przedstawiony w rozdziale *Sprzężenie absolutne osi* jest tutaj w pełni adekwatny.

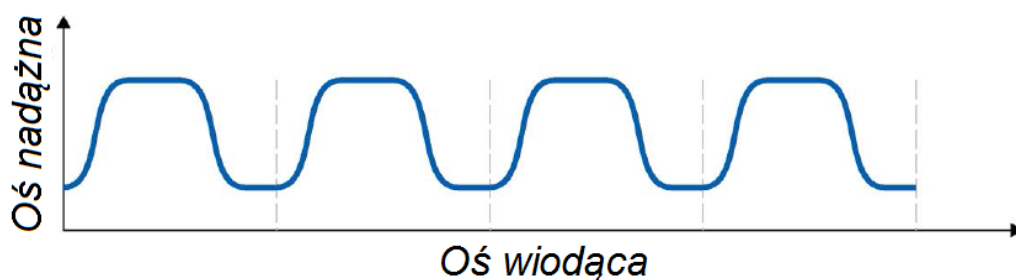
Tryb pracy

Jedyną całkowicie nową parametrem to *ApplicationMode*. Wartość tego parametru określa tryb wywołania krzywki w programie sterownika. Dostępne są trzy warianty:

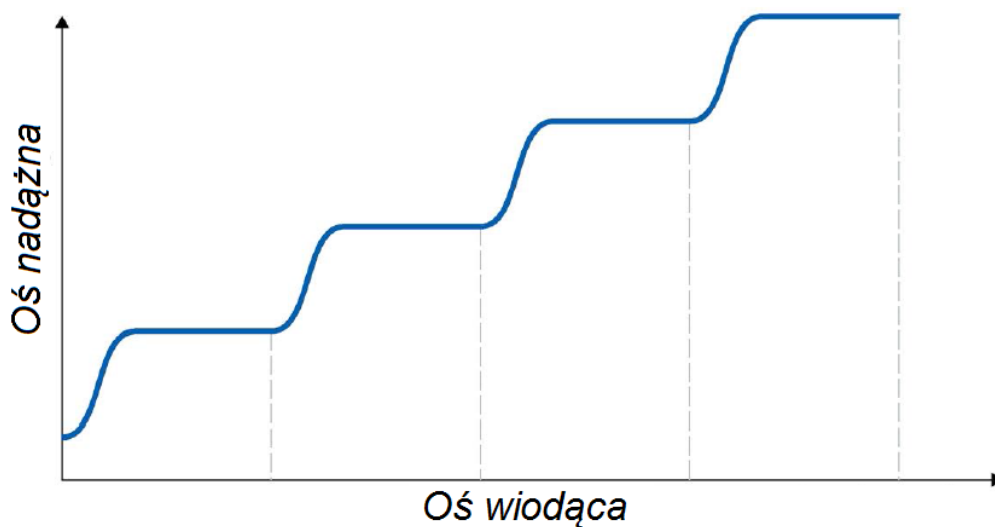
- wywołanie jednorazowe (*ApplicationMode* = 0) - ruch synchroniczny zakończony jest po osiągnięciu punktu końcowego profilu (lub początkowego jeśli wywołanie jest w kierunku przeciwnym). W celu zabezpieczenia układu przed skokowymi zmianami dynamiki – prędkość osi nadążnej musi być zerowa w skrajnych punktach profilu.



- wywołanie cykliczne (*ApplicationMode* = 1) – profil wykonywany jest rekurencyjnie, w momencie osiągnięcia punktu końcowego profilu, zostaje automatycznie powtórzony od punktu początkowego. Podobnie jak poprzednio – w celu zabezpieczenia układu przed skokowymi zmianami dynamiki – prędkość osi nadążnej musi być zerowa w skrajnych punktach profilu.



- wywołanie cykliczne dodawane (*ApplicationMode* = 2) – profil wykonywany jest rekurencyjnie, w momencie osiągnięcia punktu końcowego profilu, staje się on nowym punktem początkowym w kolejnej iteracji. W celu zabezpieczenia układu przed skokowymi zmianami dynamiki – prędkość osi nadążnej musi być ciągła w skrajnych punktach profilu.



Kierunek synchronizacji

Dla osi wiodącej o charakterystyce modulo, istotnym będzie również kierunek synchronizacji osi nadążnej. Ustawienie odpowiedniej wartości parametru *SyncDirection* pozwoli na wybór kierunku – zgodnie lub przeciwnie z przyrostem wartości, ewentualnie automatycznie w dowolnym kierunku najkrótszą ścieżką.

Parametry wyjściowe

Wyjścia funkcji *MC_CamIn* dostarczają informacji na temat statusu sprzężenia (sprężony, zajęty, procedura przerwana), rozpoczęcia oraz osiągnięcia ostatniego punktu profilu (status ustawiany przynajmniej na jeden cykl programu) oraz ewentualnych błędów.

Odsprężenie osi slave

Status sprzężenia osi synchronicznej profilu krzywkowego (*MC_CamIn.InSync* oraz *MC_CamIn.Busy*) pozostaje aktywny do momentu zakończenia pracy synchronicznej. Odsprężenie osi *slave* następuje w momencie wywołania dowolnej innej funkcji *Motion Control* na jej obiekcie tech-

nologicznym (*TO_SynchronousAxis*), np. przez zatrzymanie, bazowanie bądź zadanie prędkości czy pozycjonowanie. Błąd osi nadążnej również spowoduje zakończenie pracy synchronicznej.

20.5 Edycja krzywek w programie PLC

Na początku rozdziału wspomnieliśmy o możliwości edycji zdefiniowanych profili krzywkowych, również w trybie pracy systemu. Istotnie, w zależności wymagań aplikacji możemy zastosować różne metody pozwalające na elastyczne dostosowanie parametrów sprzężenia, do potrzeb produkcyjnych bądź uruchomieniowych.

Podczas budowania oraz testowania układu bardzo przydatna jest łatwość edycji parametrów krzywki – zarówno z poziomu środowiska inżynierskiego, jak i programu sterownika. Sterownik S7-1500T pozwoli nam na elastyczną modyfikację parametrów sprzężenia w celu usprawnienia fazy inżynierskiej projektu.

Modyfikacja projektu to jednak nie wszystko – jeszcze więcej zalet dostrzeże w otwartości systemu użytkownik

końcowy, który z łatwością będzie mógł samodzielnie dostosować tryb pracy urządzenia, nawet bez dostępu do środowiska narzędziowego oraz projektu. Modyfikacja funkcji sprzęgającej w trybie pracy systemu sterowania, daje możliwość przebrojenia linii na inny produkt, modyfikacji parametrów wytwarzanego produktu, dostosowania do zmian podzespołów lub półproduktów, kalibracji układu, zmiany punktu odniesienia osi lub ich skalowania oraz wiele innych zależnych od aplikacji funkcji, które mogą zostać wdrożone nawet przy wykorzystaniu jedynie urządzenia HMI.

Skoncentrujemy się teraz na edycji profilu w trybie pracy systemu. Rozważmy kilka wariantów, które dostępne są w sterownikach serii S7-1500T.

Styczna definicja obiektów technologicznych

Przyjmijmy, że konstruowana maszyna ma zostać przystosowana do wytwarzania wielu produktów lub jego wariantów, co wiązać się będzie bezpośrednio z koniecznością zmiany profilu sprzężenia osi skoordynowanych. Podstawową cechą systemu jest możliwość zdefiniowania wielu profili krzywkowych (tak jak wykonaliśmy to w poprzedniej sekcji), a następnie swobodna ich implementacja przez funkcje programu użytkownika. Taka metoda sprawdzi się w układach, gdzie będziemy w stanie określić zapotrzebowanie na konkretne funkcje krzywkowe, już na etapie inżynierskim.

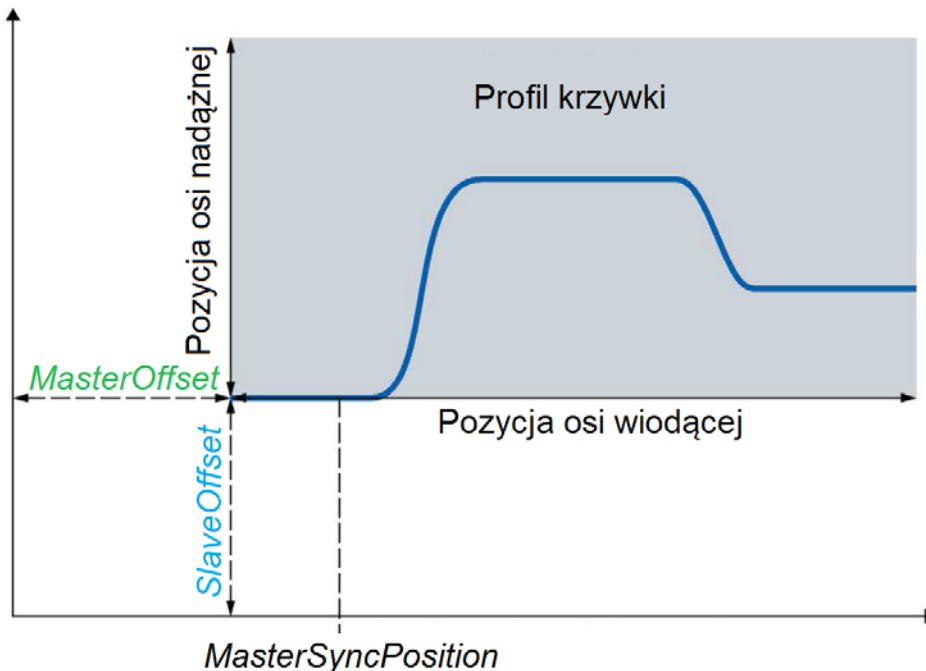
W programie sterownika trzeba jedynie wywołać odpowiednią ilość razy funkcję *MC_InterpolateCam* oraz *MC_CamIn*.

Metoda ta, przy każdej zmianie profilu, wymagać będzie interpolacji oraz aktywacji krzywki, aczkolwiek definicja wielu obiektów technologicznych *TO_Cam* nie zajmuje zasobów *Motion Control*, których liczba dla danego typu CPU jest ograniczona (*Tools*→*Resources*→*Motion Control resources*).

Przesunięcie profilu

Idąc krok dalej – założmy, że mamy zdefiniowane statycznie profile krzywek podobnie jak w powyższym przykładzie, z tym, że chcemy mieć również możliwość edycji tychże funkcji (przez program sterownika). Adaptacja profilu może odbywać się przez parametry funkcji sprzęgającej (*MC_CamIn*) lub przez modyfikację technologicznego bloku danych. Zaczniemy od podstawowych, oraz łatwych w obsłudze, opcji dostępnych z poziomu funkcji interfejsu użytkownika.

Wspomniane już parametry *MasterOffset* oraz *SlaveOffset* pozwalają na zdefiniowanie wektora przesunięcia całego profilu na płaszczyźnie zależności osi wiodącej względem nadążnej. Mówiąc inaczej możemy przesunąć punkt startowy profilu, a co za tym idzie – każdy kolejny, który został jednoznacznie wyznaczony podczas interpolacji. Sytuację ilustruje poniższa grafika.



Zwróćmy również uwagę, iż przesuwanie całość profilu – przemieszczamy także pozostałe parametry zdefiniowane dla sprzężenia, czyli np. punkt dojścia do zasprzężenia określony pozycją osi wiodącej (*MasterSyncPosition* – patrz *Synchronizacja absolutna*) czy zakres wartości osi nadążnej.

W przypadku przesunięcia wertykalnego (*SlaveOffset*) zmieniamy zakres wartości, w którym pracować będzie oś nadążna. Przykładowo przyjmijmy domyślny zakres osi 0-10 mm. Jeśli przesuniemy go o 15 mm, zakres zostanie zmieniony na 15-25mm. Trzeba tutaj zwrócić uwagę na sytuację gdzie dojście do zasprzężenia realizowane będzie „na drodze”. Zależnie od pozycji wyjściowej osi nadążnej – dystans przeznaczony na synchronizację nie ulegnie zmianie, natomiast odległość jaką będzie musiała pokonać oś nadążna, już tak.

Wykorzystując przesunięcie horyzontalne profilu (*MasterOffset*) nie zmieniamy programowego zakresu wartości osi wiodącej (np. przy osi obrotowej modułu), a jedynie przesuujemy jej charakterystykę względem osi nadążnej. Od strony dynamiki, dojście do zasprzężenia będzie realizowane więc analogicznie jak w sytuacji bez przesunięcia, z tym, że w innym punkcie osi wiodącej.

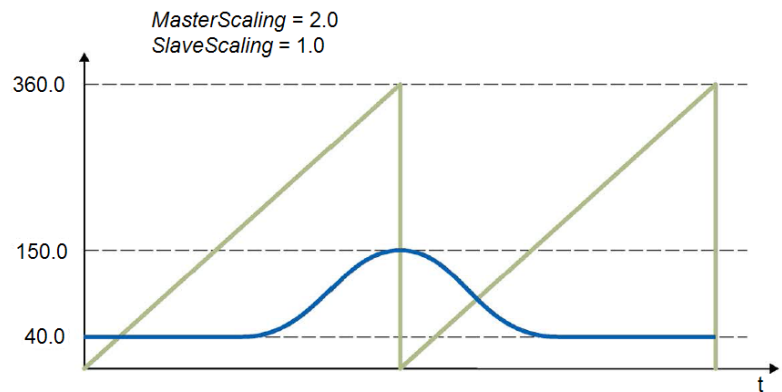
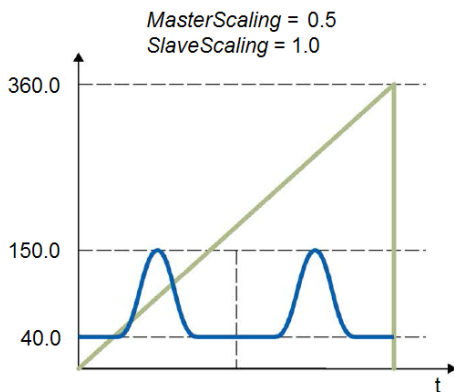
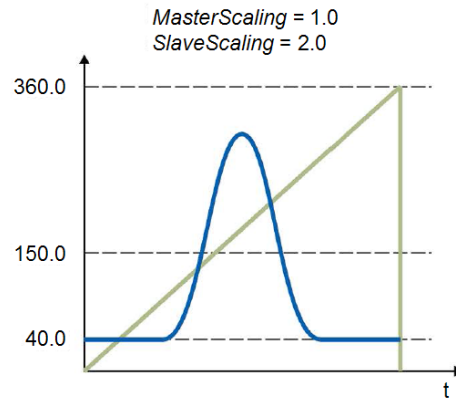
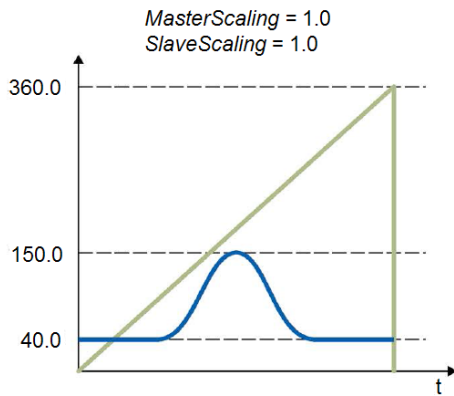
Przesunięcie profilu nie wymaga ponownej interpolacji – wystarczy podanie zbocza narastającego na wejście *Execute* funkcji *MC_CamIn*. Zmiana powyższych parametrów nie wprowadza zmian do krzywki zdefiniowanej w środowisku inżynierskim.

Skalowanie profilu

Poza możliwością przesunięcia całości zdefiniowanej funkcji, wraz z zakresem jej wartości oraz argumentów, funkcja systemowa umożliwia także zmianę „zagęszczenia” wartości względem osi układu współrzędnych. Innymi słowy możemy rozciągnąć lub skurczyć zdefiniowany profil, czyli wykonać jego skalowanie. W celu realizacji tego zagadnienia przygotowano dwa parametry funkcji *MC_CamIn*

– *MasterScaling* oraz *SlaveScaling*. Odpowiadają one za skalowanie wartości osi *master* oraz osi *slave*.

Poniższe przebiegi czasowe, ilustrują reakcję systemu na wybrane wartości parametrów skalujących zakres wartości osi *wiodącej* oraz *nadążnej*.



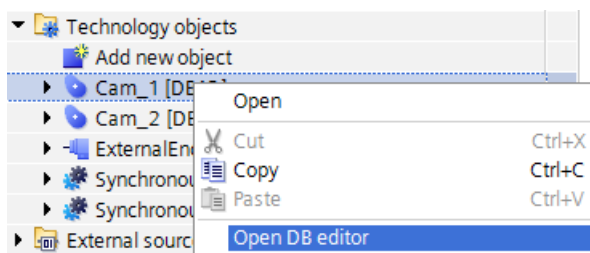
Podobnie jak w przypadku przesunięcia profilu – skalowanie powoduje zmianę zakresu pracy osi *nadążnej* lub rozkład jej wartości względem osi *wiodącej*. Nie ulegają zmianie parametry dojścia do zasprężenia (np. *MasterSyncPosition*).

Skalowanie profilu nie wymaga ponownej interpolacji – wystarczy podanie zbocza narastającego na wejście *Execute* funkcji *MC_CamIn*. Zmiana powyższych parametrów nie wprowadza zmian do krzywki zdefiniowanej w środowisku inżynierskim.

Tworzenie krzywki w trybie RT

Powyższe metody z pewnością znajdą zastosowanie w systemach, gdzie liczba profili będzie ograniczona lub różne warianty produkcyjne będą wiązać się jedynie z wektorową lub proporcjonalną modyfikacją zdefiniowanych sprzężeń. Co w takim razie w przypadku gdy zajdzie potrzeba stworzenia całkowicie nowego profilu krzywki?

Przejdźmy do najbardziej zaawansowanej oraz otwartej metody jaką możemy tutaj wykorzystać, a mianowicie do swobodnej edycji profilu przez modyfikację obiektu technologicznego z poziomu programu sterownika.



Widok tabelaryczny odzwierciedla wszelkie parametry konfiguracyjne, które ustawimy przez jego graficzny odpowiednik. W przypadku *TO_Cam* znajdziemy tutaj zarówno szczegółowe informacje na temat statusu, błędów lub ostrzeżeń generowanych przez obiekt technologiczny, jak i samą konfigurację profilu – od ustawień osi wykresu, przez parametry interpolacji, a kończąc na zdefiniowanych punktach oraz segmentach funkcji sprzęgającej.

	Name	Data type	Start value
9	ValidPoint	Array[1..1000] of Bool	
10	Segment	Array[1..50] of TO_Cam_Str..	
11	Segment[1]	TO_Cam_Struct_SegmentDa	
12	xmin	LReal	0.0
13	xmax	LReal	160.0
14	a0	LReal	0.0
15	a1	LReal	0.0625372361093122
16	a2	LReal	-7.83577256832016E-05

W przykładowym profilu – segment pierwszy zawiera się w przedziale 0.0 – 160.0 wartości osi wiodącej. Pozycja wyjściowa osi nadążnej w tym zakresie ma wartość 0.0 (a_0). Możemy też odczytać początkową wartość pochodnych w tym punkcie (a_1 , a_2 , a_3), co można skorelować z wartościami przedstawionymi w edytorze graficznym.

Wartości zawarte w bloku danych mogą nam posłużyć do graficznej reprezentacji krzywki, np. na ekranie panelu operatorskiego. Kontynuując jednak drogę do edycji profilu – możemy tutaj również wykonać modyfikację parametrów sprzężenia. Zmiana wartości rekordów bloku danych w środowisku inżynierskim, nie przyniesie jednak oczekiwanego efektu, gdyż nadrzędnym źródłem danych jest edytor graficzny. W związku z tym – wszelkie zmiany zostaną przywrócone (do ustawień początkowych) podczas kompilacji projektu.

Zacznijmy jednak od kilku zdań wstępu. Z poprzednich rozdziałów wiemy już, że każdy obiekt technologiczny, który wstawiamy do projektu jest w rzeczywistości tzw. technologicznym blokiem danych, czyli blokiem danych, który posiada zintegrowany interfejs usprawniający konfigurację, uruchomienie oraz diagnostykę osi. Pomijając jednak edytory graficzne, mamy do czynienia po prostu z blokiem danych. Łatwo zaobserwować to przez opis wstawionego do projektu obiektu technologicznego, który zawsze zawiera numer przydzielonego przez system DB. Istnieje również możliwość otwarcia widoku źródłowego technologicznego bloku danych przez wybór z menu podręcznego opcji *Open in DB editor*.

Zgodnie z wcześniejszym opisem – definicja profilu pozwala na wyznaczenie maksymalnie 1000 punktów oraz 50 segmentów opisujących relację krzywkową. Strukturę tych informacji możemy odczytać bezpośrednio z technologicznego bloku danych.

Jesteśmy w stanie jednak trwale wpływać na zawartość technologicznego bloku danych w trybie pracy aplikacji. Możemy rozważyć tutaj wariant gdzie profil mamy wstępnie zdefiniowany w środowisku inżynierskim lub utworzony został jedynie pusty obiekt technologiczny *TO_Cam*, który zostanie uzupełniony danymi podczas pracy systemu.

Modyfikacja krzywki od strony technologicznego bloku danych, nie jest zadaniem prostym. Z pewnością wymaga szerokiej wiedzy – zarówno programistycznej jak i znajomości technologii uruchamianego układu. Jest to oczywiście zadanie wykonalne aczkolwiek wychodząc naprzeciw integratorom oraz użytkownikom końcowym firma Siemens przygotowała przykład aplikacji, który w łatwy sposób można wykorzystać do zrealizowania opisywanego zadania.

Jest to gotowa biblioteka, która będzie doskonałą bazą do bezpośredniego wdrożenia w aplikacji ze sprzężeniem krzywkowym.

Szczegółowy opis zagadnienia, dokumentacja, a także przykładowy projekt można pobrać bez ograniczeń w następującej lokalizacji sieciowej:
<https://support.industry.siemens.com/cs/pl/en/view/105644659>

Opracowany przykład pozwala nie tylko swobodnie przekształcić profil krzywkowy w trybie pracy aplikacji, ale również daje możliwość wyświetlenia (np. na panelu operatorskim) aktualnie zaimplementowanego profilu.

