



LOGO! 8: pakowanie produktów a rejestry przesuwne

Rejestr przesuwany jest tematem bieżącego artykułu. Jak i w jakim celu można używać rejestru przesuwanego w Siemens LOGO! opiszę na przykładzie czterogniazdowej maszyny do pakowania produktu do pojemnika.

Opisywane urządzenie zostało w uproszczeniu przedstawione na **rysunku 1**. Składa się z obrotowego stołu oraz czterech gniazd na opakowania. Gniazdo nr 1 zawiera podajnik pojemników oraz czujnik obecności pojemnika w gnieździe. Gniazdo nr 2 jest wyposażone w dozownik produktu. Gniazdo nr 3 zawiera podajnik pokrywek i zgrzewarkę. Gniazdo nr 4 to wyrzutnik gotowych zapakowanych produktów. Co wspólnego ma czterogniazdowa pakowaczka z rejestrem przesuwnym? Aby znaleźć powiązanie musimy przeanalizować jak ona działa.

Jak działa pakowaczka

Na początku w maszynie wszystkie gniazda są puste. Zatem oprzyrządowanie w gniazdach 2, 3 i 4 nie powinno działać. Gdy w pierwszym cyklu zostanie podany pojemnik do gniazda nr 1 wówczas, po obrocie stołu o jedną czwartą obrotu, pojawi się pusty pojemnik pod dozownikiem produktu. Zatem w drugim cyklu powinien zadziałać

ponownie podajnik pojemników, aby dostarczyć kolejny kubek oraz po raz pierwszy zadziałać dozownik produktu. W trzecim cyklu, po kolejnym obrocie stołu o jedną czwartą obrotu, pojemnik z produktem pojawi się pod przyrządem nakładającym i zgrzewającym pokrywkę. Nadal jednak nie działa wyrzutnik, ponieważ pierwszy kubek do niego jeszcze nie dotarł. Co się jednak stanie, jeśli zatnie się podajnik pojemników i w jednym z cykli nie poda pojemnika, a w kolejnych cyklach będzie już normalnie podawał? Powstanie puste gniazdo, do którego podajnik produktu nie powinien nic dozować, a gniazdo nr 3 nie powinno nakładać pokrywek. Podobnie i wyrzutnik nie powinien nic wyrzucać.

Można takie zadanie rozwiązać na dwa sposoby. Pierwszy to montaż w każdym gnieździe czujników obecności pojemnika. Przy czterech gniazdach można by było to jeszcze zaakceptować. Ale jest inny sposób, tańszy, bez potrzeby kupowania i montowania dodatkowych czujników. Wystarczy zastosować rejestr przesuwany, którego zadaniem będzie przechowywanie informacji o obecności pojemnika w określonym gnieździe.

Jak działa rejestr przesuwany

Rejestr przesuwany możemy sobie wyobrazić jako rurę z otworami do zagłębienia, do której na nasz sygnał jest wkładana jedna

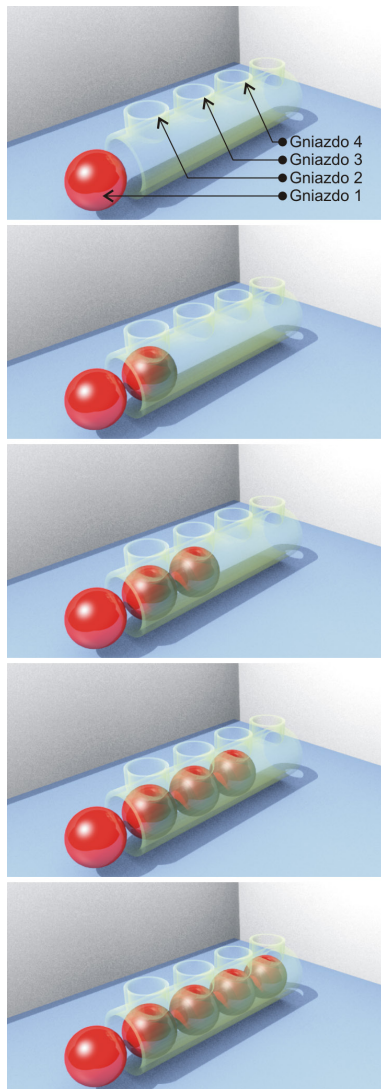
Pakowaczka czterogniazdowa



Rysunek 1. Budowa maszyny do pakowania opisanej w artykule

kula. Na **rysunku 2** pokazano naszą przykładową rurę, odwzorowującą rejestr przesuwny. Na początku rura jest pusta a, w gniazdo nr 1 wpadł pojemnik. Czerwona kula reprezentuje obecność pojemnika. W kolejnych cyklach kolejne kule są wpychane do rury. Gniazda o numerach 2, 3 i 4 obserwują przez swoje otwory czy widzą tam czerwoną kulę. Brak czerwonej kuli oznacza, że gniazdo w danym cyklu nie ma nic do zrobienia. Aby jednak kule się przesuwaly w rurze, musimy w każdym cyklu włożyć jakąś kulę. Zatem w cyklu, w którym nie ma pojemnika w gnieździe nr 1, wkładamy do rury przezroczystą kulę. Taką sytuację przedstawiono na **rysunku 3**.

Widać tutaj, że dwa cykle wcześniej nie udało się podać pojemnika do gniazda nr 1. Co dzieje się z kulami z drugiej strony rury? Wypadają z niej. W układzie rzeczywistym w LOGO! rejestr przesuwny to podwójne słowo, czyli 32

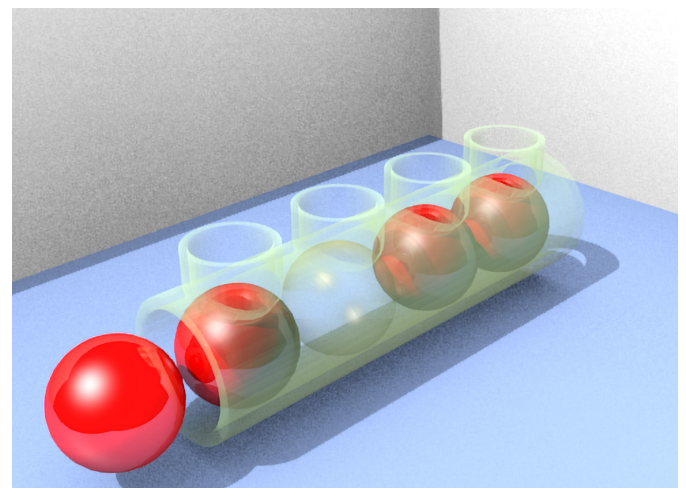


Rysunek 2. Rura ilustrująca działanie rejestru przesuwnego

bity, które możemy przesuwac na żądanie. Gdy przesuwamy bity od najniższych do najwyższych, wówczas wejście (In) rejestru podaje informację z wejścia do pierwszego bitu. Dlaczego zatem gniazdo nr 2 używa pierwszego bitu? Wynika to z faktu, że po podaniu pojemnika następuje obrót stołu, zatem informacja o pojemniku znajduje się na pozycji pierwszego bitu, ale sam pojemnik przemieszcza się na pozycję gniazda drugiego.

Praca w pętli

Na **rysunku 4** zaprezentowano program dla sterownika Siemens LOGO!, w którym użyto rejestru przesuwnego z trzema punktami dostępu do informacji. W naszym wypadku, mimo iż mamy do dyspozycji 32 bity, to używamy ich jako trzech równoległych rejestrów 8-bitowych. Każdy z elementów B001, B002 i B003 przechowuje identyczną informację, ale udostępnia inny numer bitu z tej informacji. Taki sposób dostępu do informacji wynika z faktu, że możemy przypisać jeden blok do wybranego bajtu danych w rejestrze, który składa się z 4 bajtów po 8 bitów. Zatem mamy informację o obecności pojemnika w danym gnieździe. Ale jak ona



Rysunek 3. Przezroczysta kula w rurze – ilustracja braku pojemnika w gnieździe nr 1

tam dociera? Informacja z czujnika poprzez wejście I2 jest podawana na wejścia (In) rejestrów B001, B002 i B003. Aby informacja została wpisana musimy podać sygnał na wejście (TRG), który dociera z wyjścia B008 w bloku nr 3. Jest to sygnał uruchomienia obrotów stołu.

Ruch stołu powoduje zmianę stanu wyjścia czujnika położenia stołu, które co 1/4 obrotu przyjmuje poziom niski. Sygnał z obrotu stołu jest używany do uruchomienia czynności wykonywanych we wszystkich gniazdach oraz przez wszystkie wewnętrzne układy kontrolne. Po pierwsze, w bloku nr 2 poprzez B040 kasuje pamięć wykonania zadań przez poszczególne gniazda. Kasowanie odbywa się zawsze, ale komórki pamięciowe oparte o przerzutniki RS są ponownie ustawiane, jeśli dla danego gniazda nie ma zadania do wykonania. Informacja o tym, które komórki ponownie ustawić jest pobierana z rejestru przesuwanego poprzez bramki B042, B043 i B044. Kolejnym zadaniem, które realizuje impuls z czujnika obrotu stołu to uruchomienie w bloku 7 analizy czy jest chociaż jeden pojemnik do obsłużenia. Jeśli nie będzie żadnego pojemnika, to maszyna zgłosi ich brak i wznowi swoje działanie po wyłączeniu i ponownym włączeniu przycisku START. Nasz impuls z obrotu stołu uruchamia również trzy zadania dla gniazd nr 2, 3 i 4. Uwaga, nie podaje on impulsu dla gniazda nr 1 w celu podania pojemnika. Bramki B020, B039 i B041 odpowiadają za pilnowanie, aby gniazdo wykonało pracę tylko wtedy, gdy jest pojemnik. Informacja o obecności pojemnika jest pobierana z rejestrów przesuwanych. Jeżeli wszystkie zadania zostaną wykonane, informacja o tym dotrze do bloku nr 2, który czeka na zakończenie wszystkich uruchomionych zadań. Gdy zostaną one ukończone, wyśle impuls do bloku nr 3 do bramki B006. Spowoduje to zadziałanie podajnika pojemników i rozpoczęcie kolejnego cyklu.

Zatem jakie zadania wsparliśmy stosując rejestr przesuwany i co się uprościło? Nie potrzebujemy czujników pojemników we wszystkich gniazdach, ale tylko w pierwszym. Uzyskaliśmy możliwość łatwego sprawdzania czy obsłużyliśmy już wszystkie

pojemniki i można zatrzymać maszynę. W prosty sposób uzależniliśmy wykonanie zadania od stanu bitu w rejestrze, który swoją wartością reprezentuje obecność pojemnika.

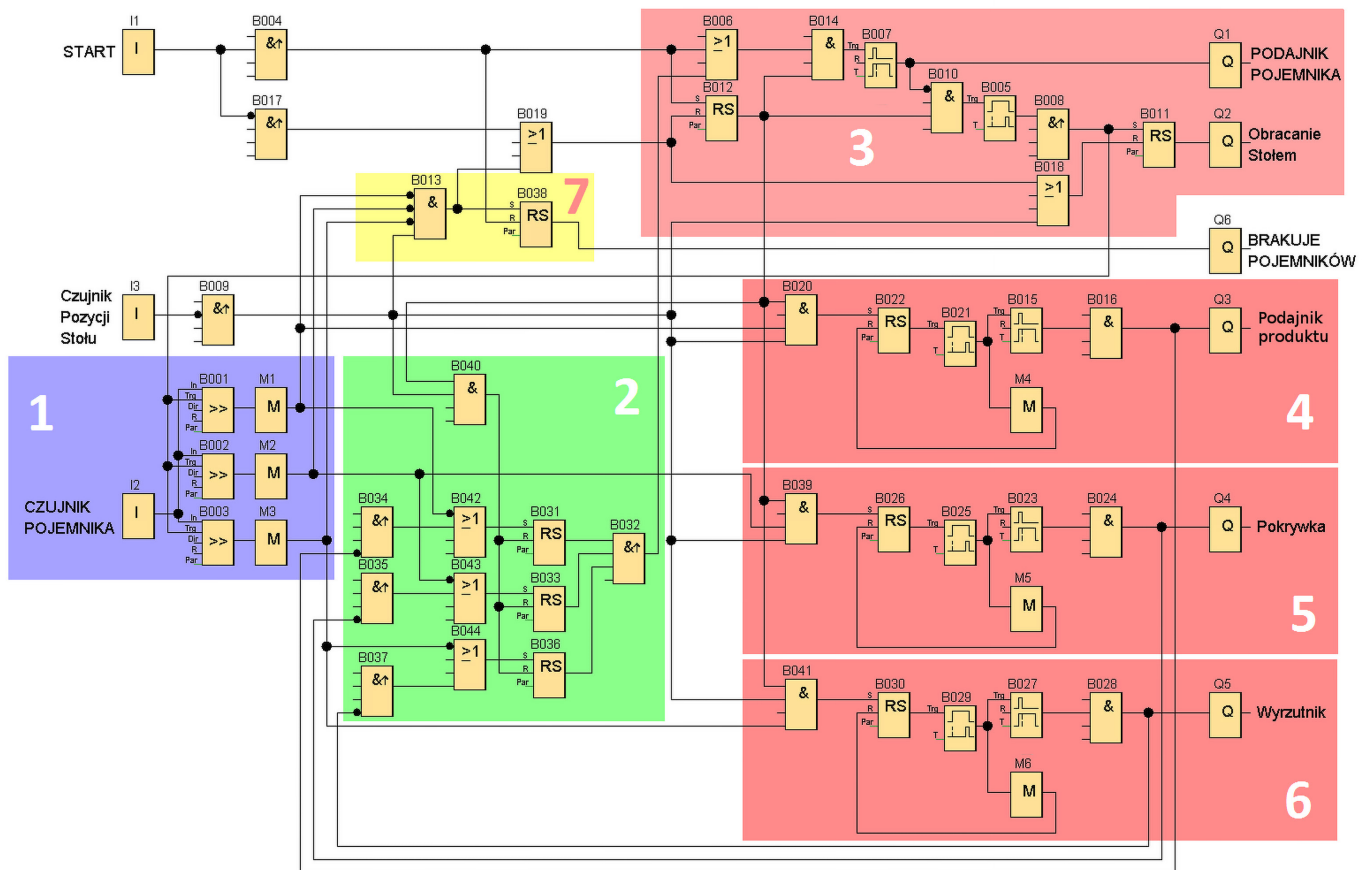
Ulepszenia

Co możemy jeszcze zmienić w naszym układzie? Pierwsze, co można zauważyć to fakt, że nowy pojemnik zostanie podany dopiero po wykonaniu zadań w gniazdach nr 2, 3 i 4. A mógłby być podawany w trakcie realizacji tych zadań. Inna opcja to dodanie możliwości powracania kubka, względem którego nie zrealizowano zadań. Dla przykładu, nie nałożono prawidłowo pokrywki i grażka nic nie zgrzała. Czujnik wychwytił ten fakt podczas obrotu stołu. Taki pojemnik mógłby wykonać dodatkową rundę, czyli wyrzutnik nie może go wypchnąć, podajnik nie może wrzucić tam pojemnika, a podajnik produktu nie może dozować porcji i za drugim razem w gnieździe pokrywek ma nastąpić druga próba nałożenia pokrywki. Analogicznie może być np. z wagą produktu, która może być realizowana pomiędzy gniazdem 2 i 3. Wówczas np. nie powinna być nakładana pokrywka.

Zarządzanie produkcją

Naszą maszynę można rozbudować. W pamięci programu pozostało nam jeszcze dużo miejsca. Możemy skomplikować proces dozowania produktu czy nakładania pokrywek lub rozbudować wyrzutnik o jakiś sorter opakowań np. po 25 sztuk. Pozostaje nam jednak interfejs operatora. Wprawdzie LOGO! ma własny ekran LCD z możliwością wprowadzania wartości parametrów, ale jednak duży monitor, a szczególnie dotykowy oraz możliwość wizualizacji stanu procesu w postaci graficznej, daje nam większe możliwości. A przecież LOGO! 8 w wersji 8.2 ma wbudowany serwer Web i obsługuje takie wizualizacje. Już niedługo poznamy tajemnice Logo Web Editor i zbudujemy stację monitorowania i zarządzania.

Arkadiusz Wernicki



Rysunek 4. Przykładowy program dla LOGO! wykorzystujący rejestr przesuwany