

4. WYKORZYSTANIE UML W PROJEKTOWANIU MECHATRONICZNYM

Przekazywanie informacji odgrywa istotną rolę w działaniu urządzeń mechatronicznych i może być łatwo przedstawione na diagramach UML (*unified modelling language*). Zdaniem autora, terminologia i notacja używana w UML może być zaadaptowana do projektowania interdyscyplinarnych systemów mechatronicznych oraz jako narzędzie do sporządzania dokumentacji na wszystkich etapach projektowania.

Komputerowe systemy CASE, stosujące język UML, są wykorzystywane nie tylko do wspomaganie pracy projektantów, ale też do zarządzania wymaganiami użytkowników i procesem projektowania. Stosowanie UML pozwala na ujednoczenie notacji, wczesne wykrywanie luk i niespójności w specyfikacji wymagań, zapewnienie zgodności użytych nazw obiektów i klas oraz ich atrybutów i metod. Użycie sformalizowanej, graficznej formy dokumentowania na wszystkich etapach projektowania ułatwia komunikowanie się członków interdyscyplinarnego zespołu przygotowującego projekt. Pozwala to na przyspieszenie prowadzonych prac oraz znacznie zwiększa szansę zakończenia sukcesem projektów dużych i złożonych systemów.

Podejście interdyscyplinarne oznacza, że konstrukcja i technologia produkcji wyrobu jest opracowana z uwzględnieniem różnych systemów i technologii, aby uzyskać efekt synergii. Członkowie interdyscyplinarnego zespołu mają różne umiejętności. Posługują się odmiennymi sposobami projektowania i opisu uzyskanych rezultatów. Daje to z jednej strony szansę stworzenia wyrobu, którego nie mógłby zaprojektować żaden z fachowców pracując oddzielnie. Z drugiej strony stwarza to potrzebę określenia wspólnego języka (terminologii i notacji), który posłuży im do precyzyjnego, łatwego i jednoznacznego porozumiewania się i dokumentowania realizowanych etapów projektu. Wspólny język powinien ułatwić precyzyjne określenie przeznaczenia, funkcji i własności projektowanego produktu i jego elementów składowych. Takim językiem może być opisany dalej UML. Jest on łatwy do zrozumienia i możliwy do zaakceptowania przez wszystkich uczestników realizujących projekt.

UML jest językiem bardzo młodym i intensywnie rozwijanym. Zdarza się, że opisy i przykłady diagramów UML w książkach i publikacjach (oraz w dostępnym oprogramowaniu) nie są zgodne z oficjalną specyfikacją tego języka. Błędy zdarzają się nawet w starannie wydanych książkach, jak wydana niedawno w Polsce monografia [18]. W tej sytuacji, w razie wątpliwości w zakresie terminologii i notacji, należy starać się sięgać do oryginalnej specyfikacji języka UML, aby

uchronić się przed możliwością powielania cudzych błędów. W niniejszej pracy opis języka UML podano w zakresie, który zdaniem autora był wystarczający do wykazania możliwości jego wykorzystania w mechatronice. Dokładniejszy opis tego języka oraz przykłady jego zastosowań można znaleźć w [7], [21], [32], [33], [64], [65], [98], [111], [113], [114] [129], [162].

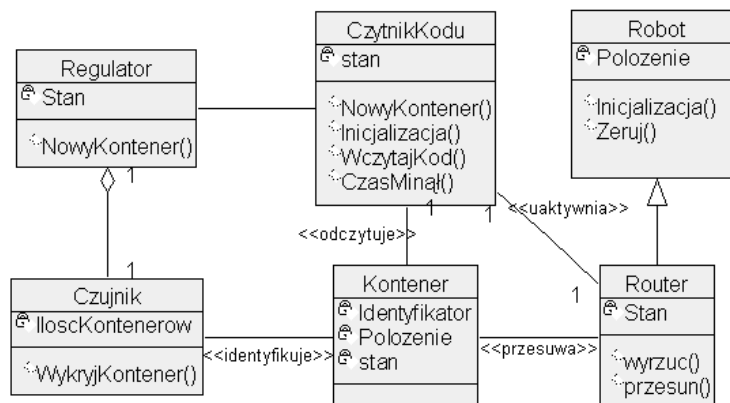
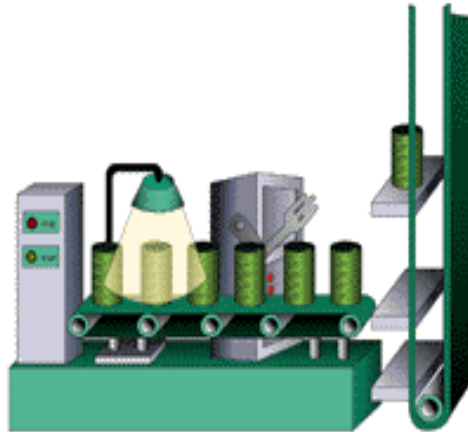
4.1. JĘZYK UML

UML stworzono w celu modelowania systemów informatycznych. Jest to język uniwersalny, obiektowo zorientowany, przydatny do projektowania systemów niezależnie od ich przeznaczenia i od języka, w którym będzie zaimplementowany docelowy system. Precyzja i uniwersalność notacji języka UML powoduje, że ponawiane są próby zastosowania go w dziedzinach innych niż informatyka. Już w roku 1998 wykorzystano UML do projektowania systemu sterowania taśmociągami i sortownikiem pojemników (rys. 4.1-1). W dalszej części rozdziału podano przykłady wybranych diagramów UML dla automatu spawającego z użyciem robota, opisywane przez autora w pracy [113].

UML jest efektem ujednoczenia trzech znanych wcześniej języków: OMT2 (*object modelling technique*) Jamesa Rumbaugh, metody Boocha oraz OOSE (*object oriented software engineering*) Ivara Jacobsona. Zawiera też pewne elementy języka SDL (*Specification and Design Language*, 1976 CCITT) oraz metodyki E-R (*entity relationship*). Rezultatem pracy dwu pierwszych autorów był UML 0.8 powstały w roku 1995 [16]. Język UML był wielokrotnie ulepszany [17] i dopiero wersja UML 1.3 została zaaprobowana w roku 1999 przez OMG (*Object Management Group*) jako propozycja standardu. Zanim to nastąpiło, UML był przedstawiony do weryfikacji w ramach obowiązującej w OMG procedury składania propozycji RFP (*request for proposal*), a następnie dyskusji (*revision task force*). Aktualnie dostępna specyfikacja to wersja 1.4 [119]. Ponadto są już przygotowywane dokumenty do języka UML w wersji 2.0.

Model projektowanego systemu ma postać diagramów graficznych. Każdy rodzaj diagramu pokazuje te elementy przyszłego systemu, które są istotne z wybranego punktu widzenia, a omija bądź upraszcza inne elementy. Wysoki poziom abstrakcji pozwala na zwięzły opis nawet bardzo dużych systemów w postaci diagramu przypadków. Jest on uzupełniany opisem werbalnym w postaci scenariuszy. Komentarze i inne informacje można umieszczać na diagramach w formie notatek, wpisywanych do specjalnej ikony. Oczywiście jest, że użycie tylko jednego typu diagramu nie jest wystarczające (rozdział 6). Praktycznie zawsze wykorzystuje się oba diagramy statyczne:

- diagram przypadków użycia (*use cases*),
- diagram klas.



Rysunek 4.1-1. Przenośnik taśmowy z sortownikami pojemników. oraz diagram klas dla tego systemu [98]

Wybór dodatkowych diagramów jest uzależniony od aktualnych potrzeb i doświadczenia projektantów. Są to **diagramy dynamiczne** (behavioralne):

- diagramy interakcji:
 - diagram sekwencyjny (w [18] używa się nazwy *diagram przebiegu*),
 - diagram współpracy (*collaboration*), zwany też diagramem czynności,
- diagram aktywności (*activity*),
- diagram stanu (*statechart*).

Diagramy implementacyjne: diagram komponentów i diagram konfiguracji (*deployment*, zwany też diagramem wdrożeniowym), nie są dostosowane do specyfiki projektowania systemów mechatronicznych i nie będą tu omawiane. Ich rolę może pełnić **diagram architektury**, przydatny w projektowaniu systemów czasu rzeczywistego. Jest on zdefiniowany w RtS (*Real-time Studio* [127]),

Diagramy i ich elementy można grupować w pakietach i w podsystemach. Pakiety i podsystemy mogą tworzyć wielopoziomą strukturę hierarchiczną, gdyż w ich skład mogą wchodzić inne pakiety i podsystemy.

Reasumując, język UML:

- stworzono do modelowania systemów informatycznych; jest uniwersalny, obiektowo zorientowany; jest przydatny do projektowania systemów niezależnie od ich przeznaczenia; model projektowanego systemu ma postać diagramów graficznych.

Możliwości języka UML zostały szybko docenione przez rynek. Pakiety CASE (*computer aided system engineering*) oferują narzędzia wspomagające kolejne etapy komputerowego wspomaganie projektowania z wykorzystaniem języka UML. Obecnie oferowane jest ponad 60 pakietów oprogramowania o różnej jakości, wspierających wykorzystanie UML. Dwa z nich: Rational Suite i Artisan Real Time Studio zostały przedstawione w tej pracy, w rozdziałach 4.5 i 4.5.3. Umożliwiają one automatyczne dokumentowanie prac i wprowadzanych zmian. CASE sprawdza też formalną poprawność i zgodność wykonywanych diagramów z diagramami wcześniej przygotowanymi. Pozwala to na automatyczne wychwytywanie błędów formalnych i braku spójności różnych elementów projektu. W rezultacie koszt i strata czasu na dokonanie poprawek i modyfikowanie projektu są relatywnie niskie, gdyż znaczną ich część wykonuje się we wczesnym etapie realizacji projektu. Stosowanie narzędzi **CASE poprawia komfort** pracy projektantów i wydaje się **niezbędne nawet przy tworzeniu niewielkich systemów**. Dodatkowym walorem jest automatyczne generowanie dokumentacji budowanego systemu przez narzędzia CASE.