PROJEKTOWANIE ARCHITEKTURY OPROGRAMOWANIA SYSTEMU STERUJĄCEGO ZROBOTYZOWNYM MODUŁEM TECHNOLOGICZNYM. CZĘŚĆ I. MODELOWANIE STRUKTURY OPROGRAMOWANIA (STATYKA)

Designing architecture software control system with a robotsizable technological module. Part I. Modeling of the software structure (static)

Jerzy STAMIROWSKI

S t r e s z c z e n i e: Konsekwencją stosowania w systemach sterowania techniki komputerowej jest przeniesienie obciążenia funkcjonalnego z węzłów mechanicznych na składniki intelektualne (elektronika, komputery). Sformułowanie wymagań, modelowanie struktury i architektury oprogramowania, należą do trudnych problemów projektowania komputerowych systemów sterowania. Jest to szczególnie istotne w dobie zintegrowanego wytwarzania. Do najbardziej popularnych narzędzi modelowania struktury i zachowania systemów komputerowych należy język UML (Unified Modeling Language). W artykule przedstawiono modelowanie struktury oprogramowania sterującego zrobotyzowanym modułem technologicznym – ZMT (ang. robotic technological module – RTM) z użyciem języka UML. Jest to pierwszy etap projektowania oprogramowania systemów komputerowych do których należy system sterowania ZTM.

S ł o w a k l u c z o w e: zrobotyzowany moduł technologiczny, system sterowania, modelowanie, struktura oprogramowania, architektura oprogramowania

A b s t r a c t: The consequence of using computer control systems is the transfer of functional load from mechanical nodes to intellectual components (electronics, computers). Formulating requirements, modeling the structure and architecture of software are among the difficult problems of designing computer control systems. This is particularly important in the era of integrated production. UML (Unified Modeling Language) is the most popular tool for modeling the structure and behavior of computer systems. The article presents the modeling of the software structure controlling the robotic technological module (ZMT) (statics) using the UML language. This is the first stage of designing software for computer systems to which the ZTM control system belongs.

K e y w o r d s: robotic technological module, control system, modeling, software structure, software architecture

Wprowadzenie

Jednym z finalnych etapów kształtowania się kompleksowej automatyzacji produkcji przemysłowej jest integracja na bazie elastycznych systemów produkcyjnych (ESP) procesów wytwarzania, technicznego przygotowania produkcji i planowania. w wydziałach a następnie w zakładzie. Prawidłowe funkcjonowanie systemu produkcyjnego zapewnia hierarchiczny system zarządzania i sterowania.

Istnieje wiele określeń ESP, a według jednego z nich [5] ESP to sterowany środkami techniki komputerowej system o różnych powiązaniach urządzeń technologicznych, elastycznych modułów produkcyjnych (EMP), zrobotyzowanych modułów technologicznych (ZMT) i zautomatyzowanego systemu technicznego przygotowania produkcji. System powinien wykazywać zdolność do automatycznego przezbrojenia przy zmianie gabarytów i asortymentu produkowanych wyrobów.

Należy podkreślić, że jest to system sterowany środkami techniki komputerowej o zwiększających się w stosunku do niego wymaganiach, wynikających z rozwoju integracji procesów produkcyjnych. Konsekwencją stosowania w sterowaniu i zarządzaniu techniki komputerowej jest przeniesienie funkcji realizowanych dotychczas przez węzły mechaniczne na układy i systemy intelektualne. Procentowy udział funkcji realizowanych w systemach przez węzły i układy mechaniczne obniżył się z 70% na początku lat 90. XX w., do 25–30% w chwili obecnej [5].

Nieodłącznym komponentem systemów sterowania jest oprogramowanie dostosowane do każdego poziomu zarządzania i sterowania. Projektowaniu oprogramowania poświęca się w ostatnim okresie coraz więcej uwagi. Wysokie wymagania i złożoność współczesnych informatycznych systemów zarządzania i sterowania wymusiły na projektantach opracowanie metod systematyzujących i usprawniających procesy projektowania. Dotyczy to w szczególności: specyfikacji wymagań, projektowania struktury (statyka), projektowania współpracy komponentów oprogramowania (dynamika) oraz projektowania architektury.

Do najbardziej popularnych narzędzi modelowania i projektowania struktury oraz zachowania się systemów komputerowych należy specjalnie w tym celu stworzony język UML (Unified Modeling Language), oferujący

projektantom diagramy pozwalające modelować różne aspekty pracy systemu [1]. Należą do nich:

Diagramy struktury: diagram przypadków użycia, diagram klas, diagram obiektów, diagram pakietów, diagram struktur połączonych.

Diagramy zachowań (dynamiki): diagram stanów (maszyny stanowej), diagram czynności, diagram sekwencji, diagram sterowania interakcją (współpracy), diagram komunikacji, diagram harmonogramowania.

Diagramy wdrożenia: diagram komponentów, diagram rozlokowania.

W procesie modelowania nie muszą być wykorzystywane wszystkie diagramy oferowane przez język UML.

O ile UML jest powszechnie stosowany w procesach projektowania systemów przetwarzających dane, nie zdobył jeszcze zadowalającej popularności wśród twórców oprogramowania systemów produkcyjnych i wbudowanych. Należy jednak podkreślić, ze język UML znajduje się w ciągłym rozwoju i występuje również w wersji dla systemów wbudowanych i produkcyjnych [2].

Komputerowy system sterowania i zarządzania pracą przedsiębiorstwa jest systemem rozproszonym z wieloma klientami i serwerami, pracującym w czasie rzeczywistym. Jednym z jego podsystemów jest zintegrowany podsystem technicznego przygotowania produkcji i sterowania pracą ZMT montażu(obróbki). Artykuł przedstawia propozycję użycia w procesie projektowania i modelowania zadań, struktury i architektury oprogramowania ZMT metodyki opartej na diagramach języka UML.

Komplet diagramów utworzonych przez analityków i projektantów systemu odgrywa w projekcie dwie role. Jest to po pierwsze podstawowe źródło informacji o strukturze i pracy oprogramowania wspomagające pracę twórców kodu programów. Po drugie jest to źródło informacji o funkcjach, strukturze i współpracy komponentów oprogramowania dla projektantów systemów produkcyjnych współpracujących z informatykami. Artykuł kierowany jest przede wszystkim do tej grupy projektantów. Zrozumienie procesu projektowania oprogramowania oraz informacji przekazywanej przez diagramy pozwala zaprojektować system o lepszej jakości i w krótszym czasie.

Modelowanie zadań oprogramowania systemu sterowania ZMT obróbki (montażu)

Zadania systemu sterowania ZMT montażu (obróbki)

Zakład produkcyjny wyposażony jest w moduł montażu (obróbki), w którym stacje robocze są usytuowane liniowo. Części przemieszczane są pomiędzy stacjami przy pomocy transportera. Na każdej stacji roboczej mają miejsce określone operacje procesu technologicznego. Dzięki temu, że stacje robocze są programowalne, możliwe jest wytwarzanie różnych wariantów gotowego produktu. Zwyczajowo detale montowane (obrabiane) są seriami.

Na stacji roboczej znajduje się robot montażowy (obróbkowy), przeznaczony do montażu (obróbki) oraz robot transportowy, który przenosi części z transportera do strefy montażu (obróbki), a po montażu (obróbce) ze strefy montażu na taśmę. Każda stacja z robotem wyposażona jest w sensory i napędy. Sensory używane są do monitoringu stanu stacji roboczej (np. rozpoznanie przybycia kolejnej części), zaś napędy – do uruchamiania i zatrzymywania mechanizmów (np. transportera). Pierwsza stacja robocza jest stacją załadunku z magazynu bufora na transporter, a ostatnia stacją wyładunku z transportera do magazynu bufora. Stacje załadunku i wyładunku wyposażone są tylko w roboty transportowe. Wszystkie stacje na których realizowane są operacje montażu(obróbki) nazwane są stacjami liniowymi. Stan stacji i stan sygnałów alarmowych kontroluje operator [5], [6].

Operacje niezbędne do wytworzenia gotowego wyrobu z materiału wejściowego przedstawione są w karcie procesu technologicznego, przygotowanej przez inżynierów-technologów. Karta określa rodzaj operacji procesu technologicznego i kolejność ich wykonania. Każda operacja realizowana jest na określonej stacji roboczej. Uruchomienie nowego procesu wytwarzania zaczyna się od utworzenia przez kierownika zakładu zamówienia zawierającego rodzaje części oraz wielkość partii.

Do podstawowych zadań systemu sterowania ZMT należy kontrola pracy i organizowanie współpracy urządzeń technologicznych, transportowych i robotów w sposób pozwalający realizować zakładane zadania produkcyjne. Znaczącą rolę odgrywa w tych procesach oprogramowanie.

Opis funkcjonowania ZTM dla potrzeb oprogramowania systemu sterowania powinien być zwięzły i dokładny. Na podstawie opisu identyfikowani są użytkownicy systemu (rzeczowniki), obiekty obszaru przedmiotowego uczestniczące w zadaniach sterowania (rzeczowniki) oraz zachowanie obiektów (czasowniki).

Zadania każdego systemu komputerowego określone są przede wszystkim potrzebami użytkowników systemu. Użytkownikami systemu mogą być ludzie i inne systemy np. systemy sterowania urządzeń. Użytkownicy mogą być użytkownikami wewnętrznymi lub zewnętrznymi.

Użytkownicy i sposób ich współpracy z systemem przedstawiany jest w języku UML na diagramach przypadków użycia (opis graficzny). Użytkownicy nazywani są aktorami, a oferowane im usługi przypadkami użycia [3], [4].

Aktorzy i przypadki użycia systemu sterowania ZMT

Diagram przypadków użycia stosowany jest do modelowania usług jakie programy systemu komputerowego świadczą aktorom, zgodnie z ich żądaniami i uprawnieniami. Nie pokazuje on na tym etapie (który jest najczęściej początkiem modelowania) konkretnych rozwiązań technicznych.

W przypadku systemu sterowania ZTM aktorami wewnętrznymi systemu są pracownicy pełniący istotne role w procesie sterowania i zarządzania. Jest to: **Kierownik Zakładu, Inżynier Technolog i Operator**.

Z systemem współpracują również aktorzy, którzy są odpowiednikami systemów zewnętrznych:**Robot Montażowy (obróbkowy) oraz Robot Transportowy**. Przypadki użycia można grupować w pakiety aktorów. O przypisaniu przypadku do pakietu decyduje inicjowanie przypadku przez aktora i udział aktora w realizacji przypadku. Poniżej przedstawiono pakiety przypadków użycia poszczególnych aktorów i należące do nich przypadki użycia

Pakiet Przypadków Użycia Operatora: *Kontrolować Sygnały Alarmowe*, *Kontrolować Stan Stanowiska Pracy, Zmienić Stan Stanowiska Pracy i Ogłosić*, *Reagować na Alarm i Ogłosić*

Np. *Reagować na Alarm i Ogłosić* Jeśli w procesie wytwarzania zaistniało zagrożenie, wszczynany jest alarm. Operator zawiadamia o tych sygnałach alarmowych do których ma uprawnienia. Jeżeli to jest możliwe usuwa przyczynę alarmu.

Pakiet Przypadków Użycia Inżyniera Technologa: *Utworzyć/Zmienić Kartę Technologiczną, Utworzyć/Zmienić Operację (extend).*

Pakiet Przypadków Użycia Kierownika Zakładu: *Utworzyć/Zmienić Zamówienie, Wyprodukować Detal.* Jako przykład, diagram użycia pakietu został przedstawiony na rys. 1.

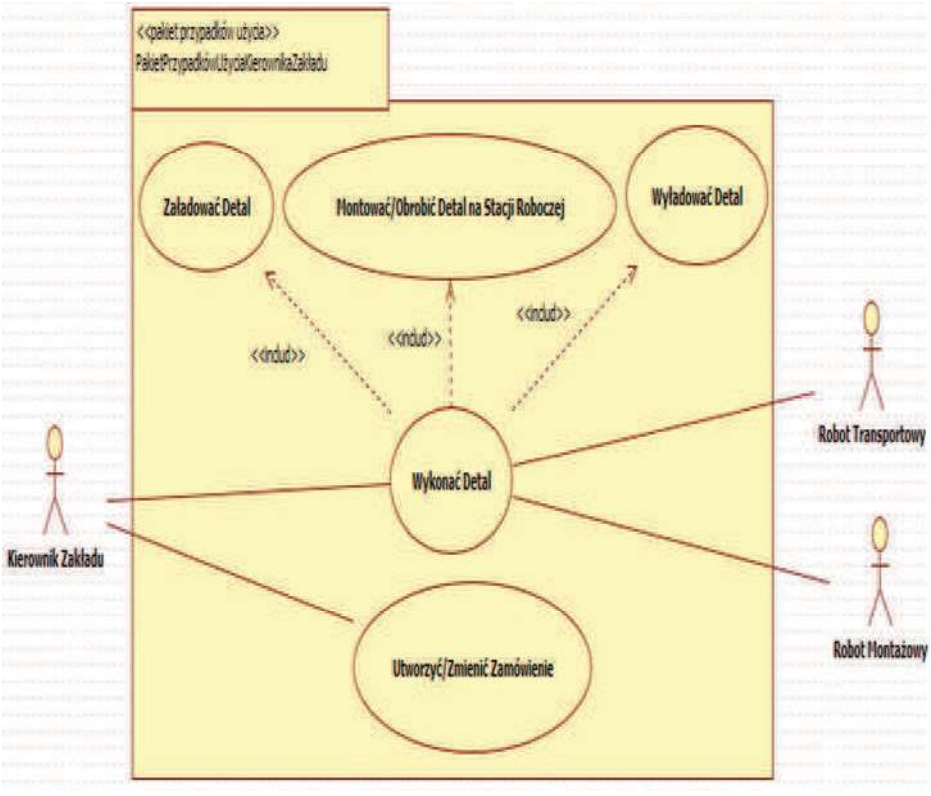
Z przypadku użycia *Wykonać Detal* można wydzielić trzy przypadki użycia: *Załadować Detal.* – dostarczenie nowego detalu na pierwszą liniową stację roboczą. *Montować(Obrobić) Detal na Stacji Roboczej* – uzyskanie detalu od stacji poprzedzającej, wykonanie operacji i przesłanie do stacji następnej (przypadek powtarza się). *Wyładować Detal.* Wysłanie gotowego detalu z ostatniej stacji liniowej do stacji wyładunku, która jednocześnie informuje kierownika o zakończeniu montażu (obróbki).

Każdy przypadek użycia realizowany jest przez odpowiednie dla jego zadań oprogramowanie.

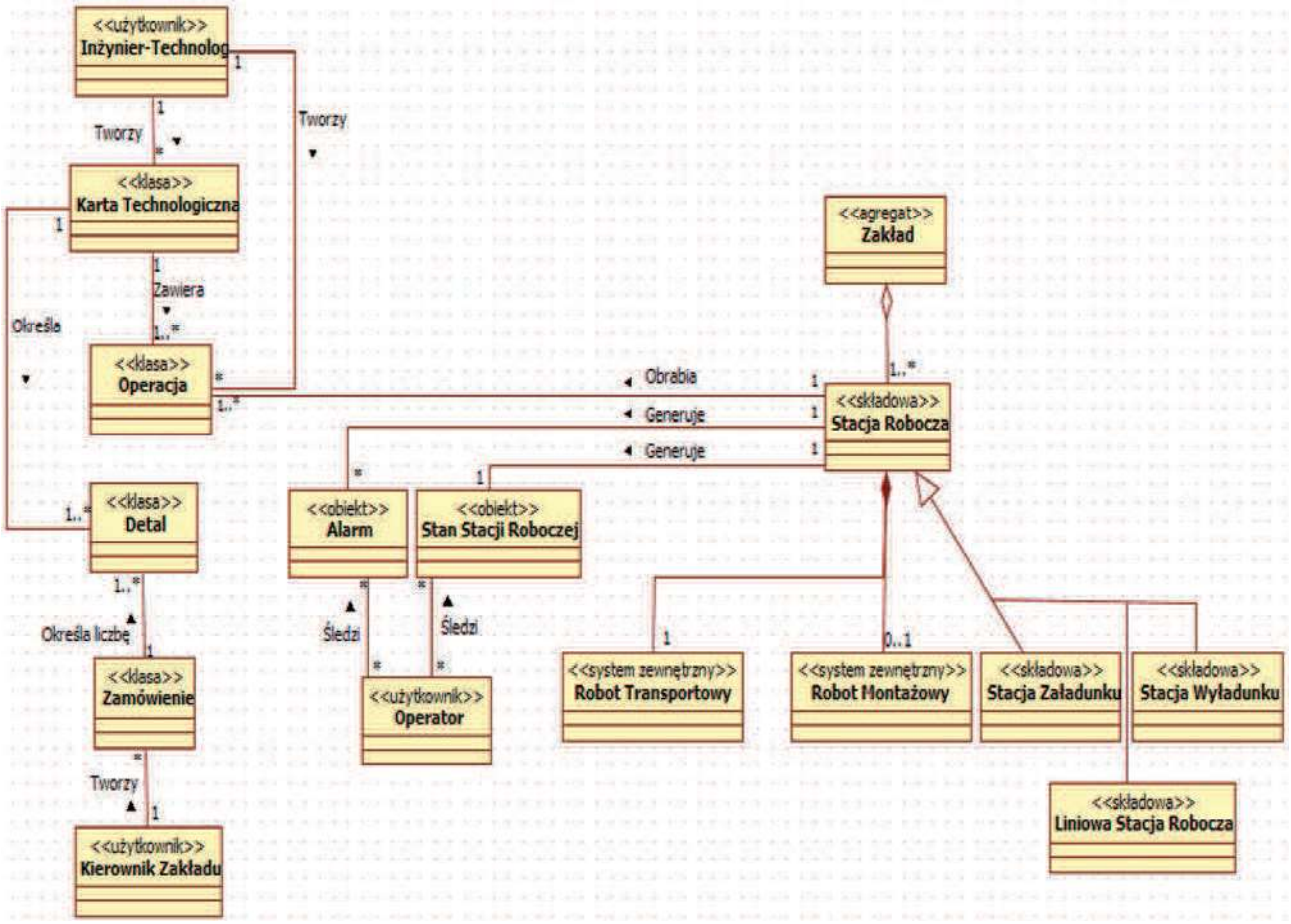
Koncepcja statycznego modelu obszaru przedmiotowego ZMT

W oparciu o analizę opisu zadania sterowania i analizę przypadków użycia specyfikuje się klasy i obiekty obszaru przedmiotowego uczestniczące w procesie sterowania (role). Oprogramowanie systemów komputerowych realizowane jest w dużej części zgodnie z paradygmatem obiektowym, dlatego model obszaru przedmiotowego powinien być budowany z uwzględnieniem zależności w jakie wchodzą obiekty obszaru przedmiotowego. Do budowy statycznego modelu obszaru przedmiotowego ZMT wykorzystano koncepcje diagramu klas języka UML.

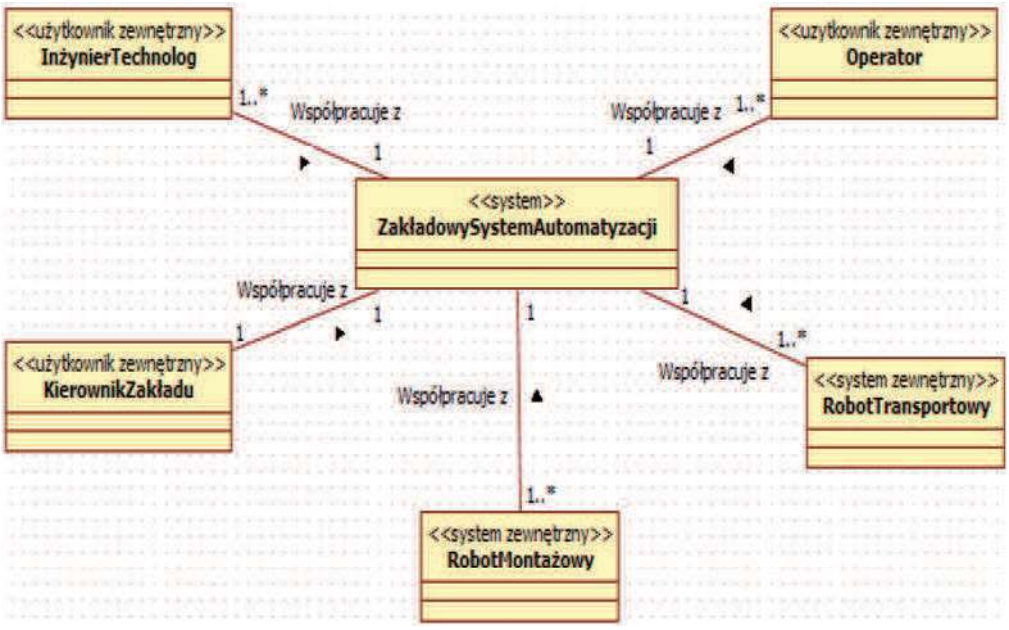
Można przyjąć, że ZMT jest klasą agregatem w skład której wchodzą klasy stacji roboczych. Wyróżnia sie trzy rodzaje stacji roboczych: załadunku, wyładunku i liniową. Stacja liniowa realizuje operacje technologiczne (montażu/obróbki). Klasy stacji wchodzą w zależności specjalizacja/generalizacja. Pomiędzy klasami obszaru przedmiotowego istnieją wielorakie związki. Klasy i związki



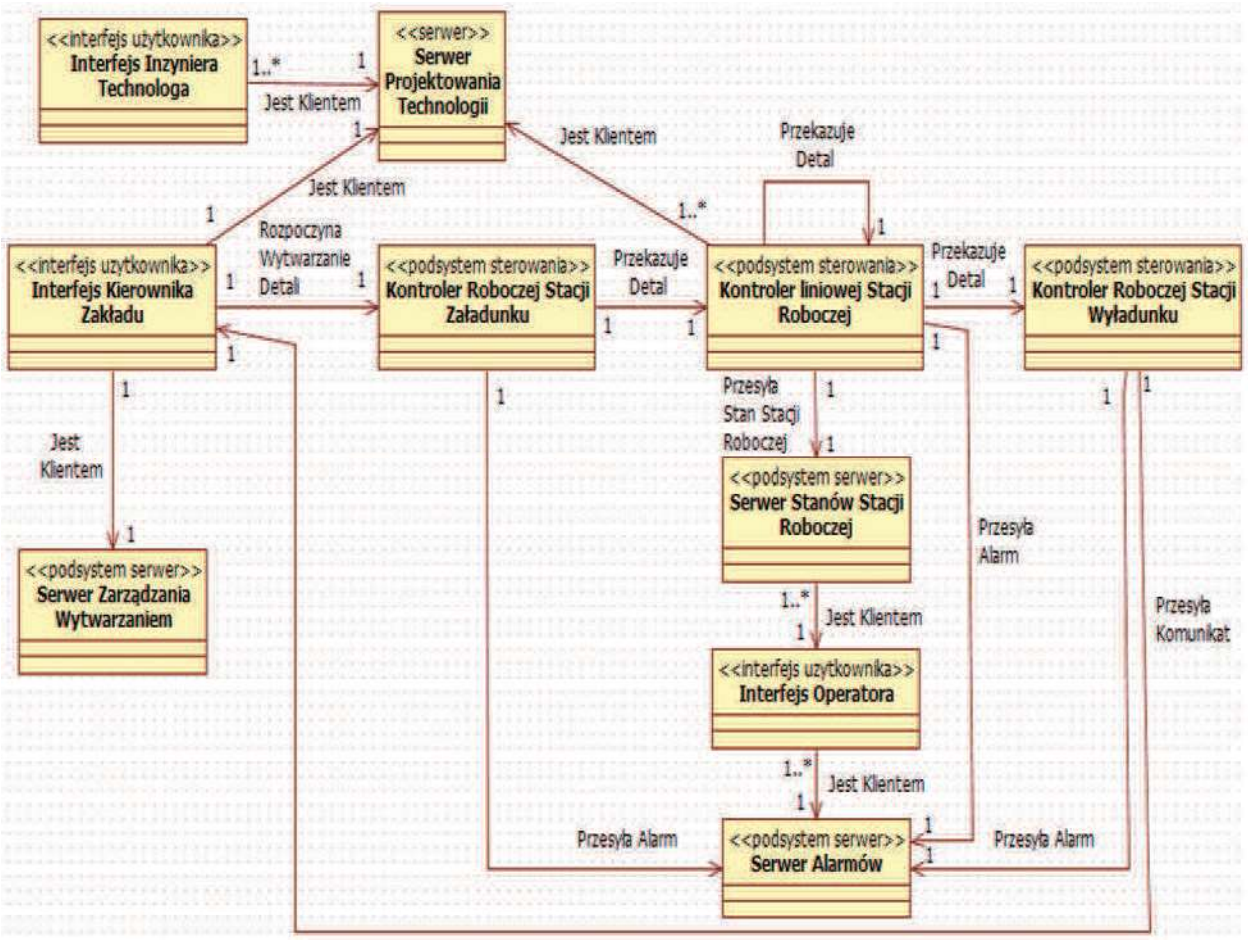
Rys. 1 Pakiet przypadków użycia Kierownika
Fig. 1 Package of Manager usage cases



Rys. 2. Model struktury obszaru przedmiotowego ZMT (statyka)
Fig. 2. The structure model of the subject area of RTM (statics)



Rys. 3. Kontekstowy diagram klas ZMT (współpraca z użytkownikami i systemami zewnętrznymi)
Fig. 3. Contextual diagram of RTM classes (cooperation with users and external systems)



Rys. 4. Klasy składowe komputerowego systemu sterowania ZMT. Model statyki oprogramowania
Fig. 4. Component classes of the RTM computer control system. The model of software statics

klas z krotnościami związków zostały przedstawione na diagramie klas obszaru przedmiotowego (rys. 2). Z każdą klasą związane są charakteryzujące klasę atrybuty.

Przedstawiony na rys. 3 kontekstowy diagram klas pokazuje współpracę systemu nadrzędnego ZMT (agregatu) z użytkownikami i systemami zewnętrznymi. Stacje robocze są obiektami wewnętrznymi dla systemu sterowania, natomiast roboty montażowy(obróbkowy) i transportowy, obsługujący stację liniową, są dla systemu sterowania systemami zewnętrznymi. Każdy z nich sterowany jest swoim układem sterowania, realizującym programy załadowane z systemu automatyzacji przedsiębiorstwa.

Klasy składowe komputerowego systemu automatyzacji wytwarzania w ZMT. Model statyki systemu komputerowego

Obiekty obszaru przedmiotowego przekształcane są w obiekty oprogramowania komputerowego systemu sterowania.

Do obiektów które stosunkowo długo przechowują informację należą: *Karta Technologiczna, Operacja, Zamówienie i Detal*. Można również zaliczyć do nich obiekty: *Stan Stacji Roboczej i Alarm*. Obiekty te są istotnymi obiektami i zostaną przekształcone na obiekty serwery do których zwracają się z żądaniem usług obiekty klienci. Na diagramie klas oprogramowania obiekty te są przedstawione jako: *Serwer Kart Technologicznych, Serwer Operacji, Serwer Zamówień, Serwer Detali, Serwer Stanów Stacji Roboczej i Serwer Sygnałów Alarmowych*. Każdy aktor-człowiek komunikuje się z systemem sterowania przy pomocy interfejsu. W systemie istnieją interfejsy: *Inżyniera Technologa, Kierownika Zakładu i Dyżurnego Operatora*. Do sterowania stacjami roboczymi wymagany jest koniecznie obiekt *Kontroler Stacji Roboczej*. W systemie istnieją: *Kontroler Stacji Załadunku (jeden), Kontroler Stacji Wyładunku (jeden), Kontroler Liniowej Stacji Roboczej (po jednym dla każdej stacji roboczej)*. Przedstawiony na rys. 4 diagram klas komponentów systemu komputerowego kończy pierwszy wstępny etap projektowania struktury oprogramowania systemu sterowania. Struktura może być uszczegółowiona po etapie modelowania zachowania się systemu (dynamiki) [3], [4].

Zamieszczone w artykule diagramy zostały wykonane przy pomocy dostępnego oprogramowania STAR UML [7]. Proponowane są również inne aplikacje o zróżnicowanych możliwościach i warunkach dystrybucji.

Wnioski i uwagi

Istniejący stan realizacji w zautomatyzowanych maszynach i systemach, szczególnie zintegrowanych, coraz większej liczby funkcji przez układy elektroniczne i systemy komputerowe (intelektualne), powinien zwrócić większą uwagę na proces projektowania oprogramowania.

Stosowanie wyspecjalizowanych narzędzi wspomagających proces projektowania oprogramowania, pozwala utworzyć komponentową (modułową) strukturę oprogramowania, skraca czas realizacji oprogramowania oraz poprawia współpracę w zespołach składających się z projektantów różnych specjalności.

Zaprojektowanie struktury oprogramowania ZMT metodą wykorzystującą diagramy języka UML pokazało, że metoda ta może być przydatna w procesie projektowania oprogramowania zautomatyzowanych maszynach i systemów.

Kolejnym bardzo istotnym etapem projektowania oprogramowania jest zbudowanie modelu współpracy komponentów oprogramowania. Etap ten jest etapem modelowana zachowania się komponentów programowych. Nazywa się go również analizą dynamiki oprogramowania. W języku UML używa się do tego celu diagramów: stanu, aktywności, sekwencji i współpracy. Etap ten pokazuje również w jaki sposób obiekty i komponenty uczestniczą w realizacji przypadków użycia a wnioski z modelowania dynamiki przyczyniają się często do wprowadzenia poprawek w strukturze oprogramowania. Kolejnym etapem jest projektowanie architektury oprogramowania, która pokazuje rozmieszczenia komponentów oprogramowania w węzłach sieci.

Diagramy tworzące projekt architektury, struktury i dynamiki oprogramowania są podstawowym i wyczerpującym źródłem wiedzy na temat funkcjonowania oprogramowania. Końcowym etapem jest napisanie z wykorzystaniem diagramów kodu programów i przetestowanie oprogramowania

Projekty struktury, dynamiki, architektury oraz przetestowany kod programów stanowią dokumentację projektową oprogramowania systemu komputerowego.

LITERATURA

[1] Craig Larman. UML i wzorce projektowe. Wydawnictwo Helion, Gliwice, 2011. s. 752.

[2] Douglass Brouce, Real-Time UML Workshop for Embedded Systems, Wydawnictwo Elsevier Science & Technology, 2014 s. 576.

[3] Gomaa Hassan, Designing Concurrent, Distributed, and Real-Time Applications with UML, Wyd. Addison-Wesley 2002.

[4] Gomaa Hassan, Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures, Cambridge University Press, New York 2011. s. 550.

[5] Monografia. Komputerowo zintegrowane projektowanie elastycznych systemów produkcyjnych. Redakcja Bogdan Palczewskyi, Antoni Świć i inni. Wydawnictwo Politechnika Lubelska. Lublin. 2015. s. 376.

[6] Zdanowicz Ryszard, Robotyzacja dyskretnych procesów produkcyjnych, Wydawnictwo Politechniki Śląskiej, Gliwice 2009. s. 427.

[7] www.staruml.io. 05.03.2018

dr hab. inż. Jerzy Stamirowski, prof. PŚk – Katedra Automatyki i Robotyki Politechniki Świętokrzyskiej, al. Tysiąclecia Państwa Polskiego 7, 25-314 Kielce, e-mail: j.stamirowski@tu.kielce.pl