

УНИВЕРЗИТЕТ У БЕОГРАДУ

МАШИНСКИ ФАКУЛТЕТ

Зоран Ж. Димић

**УПРАВЉАЧКИ СИСТЕМ ОТВОРЕНЕ
АРХИТЕКТУРЕ
РЕКОНФИГУРАБИЛНИХ РОБОТСКИХ
ЋЕЛИЈА ЗА ОБРАДУ**

докторска дисертација

Београд, 2016.

UNIVERSITY OF BELGRADE
FACULTY OF MECHANICAL ENGINEERING

Zoran Ž. Dimić

**AN OPEN ARCHITECTURE CONTROL
SYSTEM FOR RECONFIGURABLE
ROBOTIC MACHINING CELLS**

Doctoral Dissertation

Belgrade, 2016.

Комисија за преглед и одбрану:

Ментор:

Др Драган Милутиновић, редовни професор,
Универзитет у Београду, Машински факултет

Чланови Комисије:

Др Љубодраг Тановић, редовни професор,
Универзитет у Београду, Машински факултет

Др Милош Главоњић, редовни професор у пензији,
Универзитет у Београду, Машински факултет

Др Вељко Поткоњак, редовни професор,
Универзитет у Београду, Електротехнички факултет

Др Владимир Квргић, виши научни сарадник,
Лола институт, Београд

Датум одбране: _____

Mojoj porodici

ПРЕДГОВОР

Овај рад је настао као плод вишегодишњег истраживања у области развоја реконфигурабилних система управљања и програмирања обрадних ћелија на бази индустријских робота. Истраживања су спроведена током реализације више пројеката, у Лола институту, где сам запослен, и уз усавршавање на Катедри за производно машинство Машинског факултета у Београду. Експерименти, који су приказани у дисертацији, изведени су лабораторијама Катедре за производно машинство, као и у Лабораторији за машине алатке и роботе Лола института.

Како рад на дисертацији, који поред теоријских истраживања подразумева и практичну имплементацију остварених резултата, не би био могућ без помоћи институција и појединаца, као и ангажовања одговарајућих истраживачких ресурса, изражавам захвалност свима који су допринели успешном завршетку ове дисертације.

Посебну захвалност изражавам ментору проф. др Драгану Милутиновићу, редовном професору Машинског факултета у Београду за савете, подстрек и усмеравање у току израде ове дисертације.

За искрену подршку у раду, помоћ у току извођења експеримената и сугестије током израде ове дисертације, велику захвалност дугујем проф. др. Милошу Главоњићу, проф. др Саши Живановићу, доц. др Николи Славковићу. Професору др Вељку Поткоњаку изражавам захвалност на корисним саветима током израде ове дисертације.

Рад на изради дисертација не би био могућ без подршке људи из научне институције у којој радим, Лола института, па изражавам захвалност колегама на челу са директором др Владимиром Квргићем, који ми је пружио несебичну подршку.

Породици, као највећем ослоњу у животу, супрузи Драгани и деци Јелисавети и Николи, дугујем неизмерну захвалност на бризи, стрпљењу и подршци током мог научноистраживачког рада. Захваљујем се родитељима који су ми омогућили да се школујем и завршим основне студије и који су великим делом заслужни за моје професионално опредељење.

Београд, април 2016.

Зоран Димић

УПРАВЉАЧКИ СИСТЕМ ОТВОРЕНЕ АРХИТЕКТУРЕ РЕКОНФИГУРАБИЛНИХ РОБОТСКИХ ЋЕЛИЈА ЗА ОБРАДУ

АПСТРАКТ

У раду је приказан део резултата истраживања у области развоја реконфигурабилних обрадних система на бази робота за обраду делова од мекших материјала ниже и средње класе тачности, која су обухватила решавање проблема сложености програмирања робота за обраду и незадовољавајуће тачности обраде услед статичке попустљивости робота. Сложеност програмирања робота за обраду је препозната као главни ограничавајући фактор за ширу примену робота у обради резањем. Постављајући у фокус истраживања решавање овог проблема, аутор је пошао од идеје да се постојећи *CAD/CAM* системи могу користити за програмирање робота у *G*-кôду за вишеосну обраду резањем. Поставком концепта реконфигурабилне ћелије на бази робота која емулира 5-осну машину алатку и развојем метода генерализованог кинематичког моделирања обрадних ћелија на бази робота, створене су основе за развој реконфигурабилног система управљања са програмирањем обраде у *G*-кôду. На бази ових резултата постављена су два метода развоја реконфигурабилног управљачког система: (i) метод на бази примене готових хардверских и софтверских система отворене архитектуре и (ii) метод на бази примене софтверских алата и хардверских технологија отворене архитектуре уз развој алгоритама управљања. Развојем експерименталних реконфигурабилних управљачких система према постављеним методима, као и експерименталном верификацијом добијених резултата, потврђено је да се постојећи *CAD/CAM* системи могу користити за програмирање робота у *G*-кôду за вишеосну обраду резањем. Добијени резултати потврђују да је могућ развој реконфигурабилних роботских ћелија за обраду и управљачких система, који се могу реконфигурисати и од стране корисника.

Кључне речи: Роботска ћелија за обраду, моделирање робота, управљачки систем, отворена архитектура, реконфигурабилност

Научна област: Машинство

Ужа научна област: Производно машинство

УДК: 621.9-52+007.52(043.3)

004.896(043.3)

AN OPEN ARCHITECTURE CONTROL SYSTEM FOR RECONFIGURABLE ROBOTIC MACHINING CELLS

ABSTRACT

The doctoral dissertation presents part of the research in the sphere of reconfigurable robot-based machining systems for machining parts from softer materials of lower and medium accuracy grade. The research involves solving the problem of complexity of robot programming for machining and unsatisfactory machining accuracy due to robot static compliance. The complexity of robot programming for machining is identified as a major limiting factor for wider robot applications in cutting. Setting up the solution of this problem in the focus of research, the author commences from the idea that the existing CAD/CAM systems can be used for robot programming in G-code for multi-axis cutting applications. Formulating the concept of robot-based reconfigurable cell that emulates a 5-axis machine tool and developing the method of generalized kinematic modelling of robot-based machining cells, the author has created a basis for developing the reconfigurable control system with G-code machining programming. The obtained results were used to set up two methods for developing reconfigurable control system: (i) a method based on the application of ready open architecture hardware and software systems, and (ii) a method based on the application of software tools and open architecture hardware technologies, with control algorithms development. The development of experimental reconfigurable control systems according to the set up methods, as well as the experimental verification of the obtained results have confirmed that the existing CAD/CAM systems can be employed for robot programming in G-code for multi-axis cutting applications. The obtained results confirm the possibility of developing reconfigurable robotic machining cells and control systems, which can be reconfigured by the user as well.

Key words: Robotic machining cell, robot modelling, control system, open architecture, reconfigurability.

Scientific discipline: Mechanical engineering

Scientific subdiscipline: Production engineering

UDC: 621.9-52+007.52(043.3)

004.896(043.3)

Садржај:

1	УВОДНА РАЗМАТРАЊА	1
2	ВИШЕОСНА ОБРАДА РОБОТИМА.....	11
2.1	Управљање и програмирање робота за обраду.....	15
2.2	Поставка концепта реконфигурабилне ћелије за вишеосну обраду на бази робота вертикалне зглобне конфигурације	20
2.2.1	Опис концепта.....	24
3	ГЕНЕРАЛИЗОВАНИ ПРИСТУП КИНЕМАТИЧКОГ МОДЕЛИРАЊА ЕЛЕМЕНТАРНЕ ЋЕЛИЈЕ ЗА ОБРАДУ НА БАЗИ РОБОТА ВЕРТИКАЛНЕ ЗГЛОБНЕ КОНФИГУРАЦИЈЕ.....	34
3.1	Геометријски модели структура елементарних ћелија на бази робота са 5 и 6 степени слободе, које емулирају 5-осне машине алатке	34
3.2	Спољашње и унутрашње координате	39
3.2.1	Вектори унутрашњих координата	39
3.2.2	Вектори спољашњих координата.....	40
3.3	Директни кинематички проблем елементарних ћелија на бази робота као емулираних 5-осних машина алатки	45
3.3.1	Директни кинематички проблем робота вертикалне зглобне конфигурације са 5 и 6 степени слободе.....	45
3.3.2	Директни кинематички проблем елементарних ћелија, односно емулираних 5-осних машина алатки	58
3.4	Инверзни кинематички проблем робота са 5 и 6 степени слободе конфигурисаних за емулирање 5-осних машина алатки.....	65
3.4.1	Пример комплетног решења инверзног кинематичког проблема специјализованог робота вертикалне зглобне конфигурације са 5 степени слободе који емулира 5-осну вертикалну глодалицу	69
3.5	Анализа радног простора елементарних ћелија на бази робота као емулираних 5-осних машина алатки	74
4	РАЗВОЈ МЕТОДА ПРОЈЕКТОВАЊА УПРАВЉАЧКОГ СИСТЕМА ОТВОРЕНЕ АРХИТЕКТУРЕ РЕКОНФИГУРАБИЛНИХ РОБОТСКИХ ЋЕЛИЈА ЗА ОБРАДУ ..	77

4.1	Развој и имплементација реконфигурабилног управљачког система на бази готовог хардвера и софтвера отворене архитектуре применом софтвера <i>EMC2</i>	82
4.1.1	Структура софтвера <i>EMC2</i>	83
4.1.2	Систем за програмирање и симулацију реконфигурабилних обрадних ћелија на бази робота у <i>CAD/CAM</i> окружењу	88
4.1.3	Конфигурисање виртуелних реконфигурабилних ћелија за обраду на бази робота.....	91
4.1.4	Приказ морфолошког метода за конфигурисање и реконфигурисање управљачког система на бази <i>EMC2</i>	94
4.1.5	Реализација система управљања и програмирања на бази <i>EMC2</i> софтвера	99
4.2	Развој реконфигурабилног управљања на бази софтверских алата и технологија отворене архитектуре и сопствених алгоритама управљања применом <i>OROCOS</i> -а.....	101
4.2.1	Развој алгоритама и софтверских функција управљања	102
4.2.2	Поставка платформски независне структуре реконфигурабилног управљачког система.....	116
4.2.3	Имплементација модула постављеног реконфигурабилног управљачког система применом <i>OROCOS</i> -а.....	130
4.2.4	<i>Off-line</i> подсистем за програмирање и симулацију обраде	138
4.2.5	Приказ морфолошког метода за конфигурисање и реконфигурисање управљачког система на бази <i>OROCOS</i> -а	142
4.2.6	Конфигурисање експерименталног прототипа елементарне конфигурације реконфигурабилног управљачког система на бази софтверских компонената развијених применом <i>OROCOS</i> -а	150
5	ВЕРИФИКАЦИЈА ЕКСПЕРИМЕНТАЛНОГ СИСТЕМА УПРАВЉАЊА И ПРОГРАМИРАЊА РЕКОНФИГУРАБИЛНИХ ОБРАДНИХ ЋЕЛИЈА РАЗВИЈЕНОГ НА БАЗИ СОФТВЕРА <i>EMC2</i>	157
5.1	План експеримента верификације система управљања и програмирања на бази <i>EMC2</i> софтвера	157
5.2	Експерименти на роботској обрадној ћелији која емулира 5-осне вертикалне глодалице.....	158
5.3	Експерименти на роботској обрадној ћелији која емулира 5-осне хоризонталне глодалице.....	170

5.4	Експерименти примене роботске обрадне ћелије у вишестраној обради	173
5.5	Анализа реконфигурабилног управљачког система роботских ћелија за обраду развијеног на бази <i>EMC2</i>	176
6	ЕКСПЕРИМЕНТАЛНА ВЕРИФИКАЦИЈА ФУНКЦИЈА И ПЕРФОРМАНСИ ОСНОВНИХ КОМПОНЕНАТА ЗА РЕАЛИЗАЦИЈУ РЕКОНФИГУРАБИЛНОГ УПРАВЉАЧКОГ СИСТЕМА НА БАЗИ <i>OROCOS</i> -а.....	178
6.1	Верификација функција и перформанси компонената експерименталног прототипа управљачког система на тест лабораторијском систему са 3 аналогна серво погона.....	179
6.2	Верификација елементарне конфигурације експерименталног прототипа управљачког система	183
6.2.1	Анализа реконфигурабилног управљачког система на бази <i>OROCOS</i> -а.....	187
7	ЗАКЉУЧАК.....	189
8	ЛИТЕРАТУРА.....	195

Списак слика

Слика 2.1 Примери робота за обраду произвођача <i>Staubli</i> [53]	13
Слика 2.2 Примери робота за обраду произвођача <i>KUKA</i> [54]	14
Слика 2.3 Примери робота за обраду произвођача <i>ABB</i> [55]	14
Слика 2.4 Примери робота за обраду произвођача <i>FANUC</i> [56]	15
Слика 2.5 Примери комерцијалних софтверских система за програмирање и симулацију обраде роботима	17
Слика 2.6 Роботска ћелија за обраду са два робота.....	18
Слика 2.7 Експериментални стандардизовани робот вертикалне зглобне конфигурације са 6 степени слободе Лола 50.....	22
Слика 2.8 CAD модели неколико примера структура обрадних ћелија на бази расположивог стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе.....	23
Слика 2.9 Елементарна конфигурација обрадне ћелије на бази расположивог робота вертикалне зглобне конфигурације	24
Слика 2.10 Примери стандардизованог и специјализованог робота за обраду вертикалне зглобне конфигурације са 5 степени слободе.....	25
Слика 2.11 Експериментални стандардизовани робот вертикалне зглобне конфигурације са додатним интерфејсима развијеним за потербе ових истраживања	25
Слика 2.12 Петоосне машине алатке емулиране роботима вертикалне зглобне конфигурације са 5 и 6 степени слободе.....	27
Слика 2.13 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) специјализованим роботом са 5 степени слободе (конфигурација к1.1)	29
Слика 2.14 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 5 степени слободе (конфигурација к2.1)	30
Слика 2.15 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободе, $\theta_6=0^\circ$ (конфигурација к3.1)	30

Слика 2.16 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободе, $\theta_4=0^\circ$ (конфигурација к3.2)	31
Слика 2.17 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободе, при чему су све осе робота су активне ($C=0^\circ$) (конфигурација к3.3)	31
Слика 2.18 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободе, при чему су све осе робота су активне ($C=0^\circ$) (конфигурација к3.4)	32
Слика 2.19 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,C) стандардизованим роботом са 6 степени слободе, $\theta_4=\pm 90^\circ$ (конфигурација к3.5)	32
Слика 2.20 Емулирање хоризонталне и вертикалне 5-осне машине алатке, стандардизованим роботом са 6 степени слободе, приликом обраде сложене површине са једним базирањем припремка.....	33
Слика 3.1 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) специјализованим роботом са 5 степени слободе (конфигурација к1.1)	35
Слика 3.2 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 5 степени слободе (конфигурација к2.1)	36
Слика 3.3 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободе, $\theta_6=0^\circ$ (конфигурација к3.1)	36
Слика 3.4 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободе, $\theta_4=0^\circ$ (конфигурација к3.2)	37
Слика 3.5 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободе, при чему су све осе робота су активне ($C=0^\circ$) (конфигурација к3.3)	37

Слика 3.6 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободе, при чему су све осе робота су активне ($C=0^\circ$) (конфигурација к3.4)	38
Слика 3.7 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,C) стандардизованим роботом са 6 степени слободе, $\theta_4=\pm 90^\circ$ (конфигурација к3.5)	38
Слика 3.8 Придруживање координатних система сегментима манипулатора и њихови кинематички параметри	46
Слика 3.9 $D-H$ координатни системи робота вертикалне зглобне конфигурације са 6 степени слободе.....	47
Слика 3.10 $D-H$ координатни системи и кинематички параметри специјализованог робота за обраду са 5 степени.....	51
Слика 3.11 $D-H$ координатни системи и кинематички параметри стандардизованог робота за обраду са 5 степени слободе	53
Слика 3.12 $D-H$ координатни системи и кинематички параметри стандардизованог робота за обраду са 6 степени слободе и блокираном осом θ_4 ($\theta_4=\pm 90^\circ$)	56
Слика 3.13 Кинематичко декупловање.....	70
Слика 3.14 Радни простор елементарне обрадне ћелије на бази робота, која емулира 5-осну глодалицу (X,Y,Z,A,B), за случај 3-осне обраде.....	75
Слика 3.15 Радни простор елементарне обрадне ћелије на бази робота, која емулира 5-осну глодалицу (X,Y,Z,A,B), за случај 5-осне обраде.....	75
Слика 4.1 Архитектура $EMC2$	83
Слика 4.2 Метод конфигурисања $EMC2$	86
Слика 4.3 Упрошћена структура реконфигурабилног система за управљање и програмирање реконфигурабилних роботских ћелија за обраду на бази $EMC2$ софтвера.....	87
Слика 4.4 Хардвер развијеног експерименталног прототипа управљачке јединице отворене архитектуре.....	88
Слика 4.5 Упоредни приказ виртуелне обрадне ћелије на бази робота и реалног робота током обраде	89

Слика 4.6 Пример конфигурисања постпроцесора за 5-осну машину алатку конфигурације (X,Y,Z,A,B) чији рад емулира обрадна ћелија на бази робота.....	90
Слика 4.7 Конфигурисање виртуелних модела у <i>EMC2</i>	92
Слика 4.8 Виртуелни 5-осни обрадни системи на бази робота	93
Слика 4.9 <i>UML</i> дијаграм конфигурисања управљачког система на бази <i>EMC2</i>	95
Слика 4.10 Симболички приказ начина реконфигурисања	96
Слика 4.11 <i>UML</i> дијаграм реконфигурисања управљачког система на бази <i>EMC2</i>	97
Слика 4.12 Начини реконфигурисања елементарне обрадне ћелије на бази робота вертикалне зглобне конфигурације са 6 степени слоборе.....	98
Слика 4.13 Елементарна обрадна ћелија на бази робота Лола 50 са реализованим управљачким системом применом софтвера <i>EMC2</i>	100
Слика 4.14 <i>UML</i> дијаграм софтверских модула <i>NC</i> компилатора.....	103
Слика 4.15 <i>UML</i> дијаграм активности <i>NC</i> компилатора.....	105
Слика 4.16. <i>UML</i> дијаграм класа <i>Lexical Analyser</i> -а.....	105
Слика 4.17 <i>UML</i> дијаграм <i>Hash</i> -а.....	105
Слика 4.18 <i>UML</i> дијаграм класа <i>Symbol Table</i>	105
Слика 4.19 <i>UML</i> дијаграм <i>Syntax Analyser</i> -а	106
Слика 4.20 <i>UML</i> дијаграм софтверских модула <i>BNF</i> -а.....	106
Слика 4.21 <i>UML</i> дијаграм класа <i>Error Handler</i> -а	107
Слика 4.22 <i>UML</i> дијаграм активности општег интерполатора	109
Слика 4.23 Апроксимативно кретање.....	110
Слика 4.24 <i>UML</i> дијаграм активности општег кинематичког модула.....	114
Слика 4.25 Униформни интерфејс кинематичких модула	114
Слика 4.26 Архитектура <i>PCI</i> подсистема.....	115
Слика 4.27 Генеричка хијерархијска структура предложеног реконфигурабилног управљачког система	117
Слика 4.28 Градивне софтверске компоненте <i>OROCOS</i> -а [79].....	121
Слика 4.29 <i>OROCOS</i> окружење за развој <i>real-time</i> апликација	123
Слика 4.30 Структура <i>RTT</i> -а	124

Слика 4.31 Интерфејс <i>OROCOS</i> апликативне компоненте (процеса)	125
Слика 4.32 Активности које покрећу <i>TaskContext</i>	126
Слика 4.33 Дијаграм стања <i>TaskContext</i> -а	128
Слика 4.34 Апликативни интерфејс <i>OROCOS</i> компоненте.....	130
Слика 4.35 Компонента <i>generatorPos</i> - Интерполатор са интерпретером <i>P</i> кода	131
Слика 4.36 Компонента <i>kinematics</i> - Имплементација функција инверзне и директне кинематике	132
Слика 4.37 Компонента <i>nAxesContorllerPos</i> - Имплементација серво контроле позиција оса.....	133
Слика 4.38 Компонента <i>robotHardware</i> - Апликативни интерфејс ка спрежним картицама и актуаторима	134
Слика 4.39 Компонента <i>manualMode</i> - омогућава ручно управљање осам роботске ћелије.....	135
Слика 4.40 Компонента <i>initRobot</i> - омогућава иницијализацију оса роботске ћелије.....	136
Слика 4.41 Дијаграм активности функције <i>updateHook</i> садржане у компоненти <i>initRobot</i>	137
Слика 4.42 Едитор програмског кода <i>off-line</i> система.....	139
Слика 4.43 Конфигурациони интерфејс	139
Слика 4.44 Виртуелна ћелија са два робота.....	140
Слика 4.45 Комуникација између <i>real-time</i> и <i>off-line</i> подсистема	141
Слика 4.46 Иницијално конфигурисање <i>real-time</i> управљачког подсистема	144
Слика 4.47 Иницијално конфигурисање <i>off-line</i> подсистема.....	145
Слика 4.48 Пример статичког конфигурисања <i>off-line</i> подсистема, компилације и симулације програма обраде.....	147
Слика 4.49 Пример статичког реконфигурисање <i>real-time</i> управљачког подсистема.....	148
Слика 4.50 Пример динамичког реконфигурисање <i>real-time</i> управљачког подсистема.....	150
Слика 4.51 Имплементација класа <i>RTT::Motenc_Lite_Board</i> и <i>RTT::MotencEncoderLite</i>	151

Слика 4.52 Имплементација класа <i>RTT:MotencDigitalOutInterface</i> и <i>RTT:MotencDigitalInInterface</i>	152
Слика 4.53 Управљачки систем конфигуриран за управљање роботом Лола	15
.....	153
Слика 4.54 Орман управљачке јединице	155
Слика 4.55 <i>Watchdog</i> тајмер	155
Слика 4.56 Алгоритам функционисања <i>watchdog</i> тајмера.....	156
Слика 5.1 Цртеж и <i>CAD</i> модел првог пробног дела	160
Слика 5.2 Симулације путање алата и уклањања материјала за пример обраде првог пробног дела.....	161
Слика 5.3 Одређивање нулте тачке	161
Слика 5.4 Обрада првог пробног дела емулираном 5-осном вертикалном глодалицом (<i>X,Y,Z,A,B</i>) конфигурације к3.1	162
Слика 5.5 Симулације путање алата за пример обраде другог пробног дела	163
Слика 5.6 Обрада другог пробног дела емулираном 5-осном вертикалном глодалицом (<i>X,Y,Z,A,B</i>) конфигурације к3.3	163
Слика 5.7 Обрада трећег пробног дела емулираном 5-осном вертикалном глодалицом (<i>X,Y,Z,A,B</i>) конфигурације к3.1	164
Слика 5.8 Обрада импелера емулираном 5-осном вертикалном глодалицом (<i>X,Y,Z,A,B</i>) конфигурације к3.1	165
Слика 5.9 Обрада лопатице емулираном 5-осном вертикалном глодалицом (<i>X,Y,Z,A,B</i>) конфигурације к3.3	166
Слика 5.10 Виртуелна обрада анфаса људске главе у управљачком систему	167
Слика 5.11 Обрада анфаса људске главе емулираном 5-осном вертикалном глодалицом (<i>X,Y,Z,A,B</i>) конфигурације к3.1	167
Слика 5.12 Обрада профила људске главе емулираном 5-осном вертикалном глодалицом (<i>X,Y,Z,A,B</i>) конфигурације к3.1	168
Слика 5.13 Приказ путање алата и припремка за обраду рељефа грчке античке реплика.....	169

Слика 5.14 Обрада рељефа грчке античке реплике емулираном 5-осном вертикалном глодалицом (X,Y,Z,A,B) конфигурације к3.1	169
Слика 5.15 Обрада рељефа српског средњевековног грба емулираном 5-осном вертикалном глодалицом (X,Y,Z,A,B) конфигурације к3.1	170
Слика 5.16. Емулација хоризонталне глодалице конфигурације (X,Y,Z,A,B) ..	171
Слика 5.17 Симулације путање алата за пример обраде пробног дела на емулираној 5-осној хоризонталној глодалици (X,Y,Z,A,B)	171
Слика 5.18 Симулација обраде пробног дела у управљачком систему	172
Слика 5.19. Обрада пробног дела на емулираној 5-осној хоризонталној глодалици (X,Y,Z,A,B) у конфигурацији к3.4	172
Слика 5.20. Обрада анфаса људске главе на емулираној 5-осној хоризонталној глодалици (X,Y,Z,A,B) у конфигурацији к3.2	173
Слика 5.21 <i>CAD</i> модел тест радног предмета за вишестрану обраду	174
Слика 5.22 Вишестрана обрада радног предмета.....	175
Слика 6.1 Тест лабораторијски систем са 3 аналогна серво погона	179
Слика 6.2 Окружење за тестирање управљачког система.....	180
Слика 6.3 Програм на <i>L-IRL</i> језику за тестирање трајекторија три осе, θ_1 до θ_3 , у синхронном <i>PTP</i> моду.....	181
Слика 6.4 Позиције ротора мотора током тестирања драјверског модула спрежног хардвера за улазно/излазне операције	181
Слика 6.5 Робот вертикалне зглобне конфигурације Лола 15 са 6 степени слободе	183
Слика 6.6 Виртуелно <i>3D</i> окружење за тестирање програма	185
Слика 6.7 Систем за контролу и праћење рада робота по извршеном задатку	186
Слика 6.8 Експеримент на роботу Лола 15	186

Списак табела

Табела 1.1 Хијерархијски нивои софтвера управљачког система	5
Табела 3.1. <i>D-H</i> кинематички параметри	48
Табела 4.1 Структуре управљачких система машина алатки и робота	78
Табела 4.2 <i>BNF</i> нотација за основни скуп инструкција <i>G</i> -кôда.....	104
Табела 4.3 Пример компилације једне <i>G</i> - кôд линије	108
Табела 5.1 План експеримента за обраду радних предмета.....	159
Табела 6.1 Времêна извршавања припремних функција [<i>ms</i>]	182
Табела 6.2 Времêна извршавања функција интерполатора [<i>ms</i>].....	182

Номенклатура

У оквиру рада су због краћег записивања коришћене скраћенице за термине чија су значења дата у наставку.

Акроними

<i>ADC</i>	<i>Analog-to-Digital Converter</i>
<i>API</i>	<i>Application Programming Interface</i>
<i>ASCII</i>	<i>American Standard Code for Information Interchange</i>
<i>BFL</i>	<i>Bayesian Filtering Library</i>
<i>BNF</i>	<i>Backus-Naur Form</i>
<i>CAD</i>	<i>Computer Aided Design</i>
<i>CAM</i>	<i>Computer Aided Manufacturing</i>
<i>CIRP</i>	<i>College International pour la Recherche en Productique</i>
<i>CLF</i>	<i>Cutter Location File</i>
<i>CMFs</i>	<i>Canonical Machining Functions</i>
<i>CNC</i>	<i>Computer (or computerized) Numerical Control</i>
<i>COMEDI</i>	<i>COontrol and Measurement Device Interface</i>
<i>CORBA</i>	<i>Common Object Request Broker Architecture</i>
<i>COTS</i>	<i>Commercially Available Off-The-Shelf</i>
<i>CPI</i>	<i>Continuous Path Interpolation</i>
<i>DAC</i>	<i>Digital-to-Analog Converter</i>
<i>DAE</i>	<i>Differential Algebraic Equation</i>
<i>DCOM</i>	<i>Distributed Component Object Model</i>
<i>DIN</i>	<i>Deutsches Institut für Normung</i>
<i>DXF</i>	<i>Drawing Exchange Format</i>
<i>EIA</i>	<i>Electronic Industries Alliance</i>
<i>EMC2</i>	<i>Enhanced Machine Controller верзија 2</i>
<i>EMCMOT</i>	<i>EMC MOTion - EMC2 модул за управљање кретањем</i>
<i>FSM</i>	<i>Finite State Machine</i>
<i>GIOP</i>	<i>General Inter-ORB Protocol</i>
<i>GUI</i>	<i>Graphical User Interface</i>
<i>HAL</i>	<i>Hardware Abstraction Layer</i>
<i>I/O</i>	<i>Input/Output</i>
<i>IDL</i>	<i>Interface Description Language</i>
<i>IEEE</i>	<i>Institute of Electrical and Electronics Engineers</i>

<i>IFR</i>	<i>International Federation of Robotics</i>
<i>IIOIP</i>	<i>Internet Inter-Orb Protocol</i>
<i>ISO</i>	<i>The International Organization for Standardization</i>
<i>JOP</i>	<i>Japanese Open Promotion Group</i>
<i>KDL</i>	<i>Kinematics and Dynamics Library</i>
<i>L-IRL</i>	<i>Lola-Industrial Robot Language</i>
<i>LL(1)</i>	Силазно парсирање при коме се користи један <i>lookahead</i> токен за доношење одлука
<i>NC</i>	<i>Numerical Control</i>
<i>NIST</i>	<i>National Institute of Standards and Technology</i>
<i>OAC</i>	<i>Open Architecture Control</i>
<i>OCL</i>	<i>OROCOS Component Library</i>
<i>ODE</i>	<i>Ordinary Differential Equation</i>
<i>OMAC</i>	<i>Open Modular Architecture Controllers</i>
<i>OOP</i>	<i>Object Oriented Programming</i>
<i>OpenGL</i>	<i>Open Graphics Library</i>
<i>OROCOS</i>	<i>Open RObot COntrol Software</i>
<i>OSACA</i>	<i>Open System Architecture for Controls within Automation System</i>
<i>PC</i>	<i>Personal Computer</i>
<i>PCI</i>	<i>Peripheral Component Interconnect</i>
<i>PID</i>	<i>Proportional-Integral-Derivative</i>
<i>PLC</i>	<i>Programmable Logic Controller</i>
<i>PSoC</i>	<i>Programmable System-on-Chip</i>
<i>RCS</i>	<i>Real-Time Control System</i>
<i>RMAC</i>	<i>Reconfigurable Mechanisms for Application Control</i>
<i>RTAI</i>	<i>RealTime Application Interface</i>
<i>RTT</i>	<i>Real-Time Toolkit</i>
<i>ShM</i>	<i>Shared Memories</i>
<i>SSD</i>	<i>Solid-State Drive</i>
<i>STC-M</i>	<i>Scientific Technical Committee "M"</i>
<i>SWHR</i>	<i>Software-Hardware Reconfigurability</i> - софтверск-хардверска реконфигурабилност
<i>SWR</i>	<i>Software Reconfigurability</i> - софтверска реконфигурабилност
<i>TAO</i>	<i>The Ace Orb</i>
<i>TCP/IP</i>	<i>Transmission Control Protocol/Internet Protocol</i>
<i>UML</i>	<i>Unified Modeling Language™</i>

VRML	Virtual Reality Modeling Language
XML	eXtensible Markup Language

Ознаке

$\{M\}$	координатни систем основе емулиране машине алатке
$\{T\}$	координатни систем алата
T	врх глодала
$\{W\}$	координатни систем радног комада
${}^M\mathbf{v}$	вектори референцирани у координатном системиму $\{M\}$
${}^W\mathbf{v}$	вектори референцирани у координатном системиму $\{W\}$
$\boldsymbol{\theta}$	$[\theta_i]^T, i = 1, 2, \dots, n$ - вектор унутрашњих координата
${}^W\mathbf{p}_T$	$[x_{TW} \ y_{TW} \ z_{TW}]^T$ - вектор позиције врха алата T у координатном систему радног комада $\{W\}$
${}^W\mathbf{k}_T$	$[k_{TWx} \ k_{TWy} \ k_{TWz}]^T$ - јединични вектор алата у координатном систему радног комада $\{W\}$
${}^M\mathbf{k}_T$	$[k_{Tx} \ k_{Ty} \ k_{Tz}]^T$ - јединични вектор алата у координатном систему робота $\{M\}$
${}^M\mathbf{p}_{OW}$	$[x_{OW} \ y_{OW} \ z_{OW}]^T$ - позиција координатног система радног комада $\{W\}$ у координатном систему робота $\{M\}$
${}^M{}_W\mathbf{R}$	оријентација координатног система радног комада $\{W\}$ у координатном систему робота $\{M\}$
${}^M\mathbf{p}_T$	позиција координатног система $\{T\}$ у односу на координатни систем $\{M\}$
${}^M{}_T\mathbf{R}$	оријентација координатног система $\{T\}$ у односу на координатни систем $\{M\}$
z_T	оса алата
X_M, Y_M, Z_M	осе координатног система основе емулиране машине алатке
(A, B, C)	програмирани углови оријентације алата добијени из G -кôда
(X, Y, Z)	програмиране координате позиције алата добијене из G -кôда
$\mathbf{R}_{X_M, A}$	ротациона матрица - ротација алата око X_M осе за програмирани угао A
$\mathbf{R}_{Y_M, B}$	ротациона матрица - ротација алата око Y_M осе за програмирани угао B
$\mathbf{R}_{Z_M, C}$	ротациона матрица - ротација алата око Z_M осе за програмирани угао C
${}^M{}_T\mathbf{T}$	хомогена матрица - позиција и оријентација координатног система алата $\{T\}$ у односу на координатни систем основе робота $\{M\}$
\mathbf{A}	${}^{i-1}_i\mathbf{A}, i = 1, 2, \dots, n$ - D - H трансформационе матрице
${}^0_{IP}\mathbf{T}$	позиција и оријентација додатног интерфејса робота у односу на координатни систем основе робота

${}^n_{IP}\mathbf{T}$	позиција и оријентација додатног интерфејса робота у односу на координатни систем последње осе робота
${}^0_{IP}\mathbf{p}$	позиција координатног система $\{IP\}$ у односу на координатни систем основе робота
${}^0_{IP}\mathbf{R}$	оријентација координатног система $\{IP\}$ у односу на координатни систем основе робота

1 УВОДНА РАЗМАТРАЊА

Стални и све бржи развој нових производа, наметнут оштром конкуренцијом у индустријској производњи, пред произвођаче машина алатки поставља све теже и врло опречне захтеве. Са једне стране се захтева обрада делова врло сложене геометрија у малим серијама или чак уникатној производњи, док се са друге стране захтевају висока тачност и квалитет обраде, висока продуктивност и нижа цена. У циљу одговора на сложене захтеве тржишта, интензивирани су истраживања у области обрадних система, која су снажно подржана од стране произвођача машина алатки и пратеће производне опреме.

Актуелна истраживања у домену обрадних система, поред развоја 5-осних машина алатки, иду и у правцу развоја мултифункционалних и реконфигурабилних обрадних система. Међународна академија за истраживања у области производног машинства – *CIRP (College International pour la Recherche en Productique)*, са седиштем у Паризу, је у оквиру *STC-M (Scientific Technical Committee "M")* Научно техничког комитета за машине, кроз два уводна реферата [1, 2], дала пресеке стања и правце истраживања у области реконфигурабилних обрадних система и виртуелних машина алатки. Концепт мултифункционалних машина укључује велики број оса, могућност извођења различитих метода обраде резањем, мерење и контролу делова, аутоматску измену алата па чак и ласерске методе локалне термичке обраде и сл [3]. Основна идеја је да се сложени делови из дате класе комплетно обраде на једној машини и у једном постављању (енг. *single-hit machining*). Предложени концепт је заснован на великом броју управљаних оса машине, које у различитим комбинацијама конфигуришу систем за дату вишеосну обраду. Петоосне машине алатке, за обраду делова сложене геометрије и високе тачности, због високе цене и ограничених димензија радног простора није рентабилно користити за обраду делова већих димензија, од мекших материјала, са сложеним функционалним и естетским површинама и ниже класе тачности. Једну од могућих алтернатива 5-осним машинама алаткама за ову класу задатака представљају индустријски

роботи, чија је цена вишеструко нижа.

Уз особине флексибилности, програмабилности и мултифункционалности, нижа цена робота представља јак мотив за њихову примену у обради резањем уместо 5-осних машина алатки. Европски истраживачки програми, попут Оквирних програма 6 и 7 (*FP6, FP7*) и Хоризонт 2020 (*H2020*), активирали су снажне подстицаје примени робота у готово свим сегментима индустрије. Акценти су стављени на кооперативан рад човека и робота, као и на алтернативну примену робота у технолошким процесима са скупим машинама, каква је 5-осна обрада резањем делова сложене геометрије. Неколико актуелних пројеката у свету, покренутих у домену производних технологија, вишеосну обраду роботима препознаје као одговарајућу алтернативу обради 5-осним машинама алаткама [4-9]. Обрада роботима радних предмета од мекших материјала, као што су дрво, пластика, алуминијум и сл., дала је задовољавајуће резултате, што показују актуелна истраживања [10-12].

Од уобичајених конфигурација робота дефинисаних *ISO 8373* стандардом, поред Декартове конфигурације, која се по својој структури може сматрати 5-осном порталном (*Gantry*) глодалицом, као нпр. *HSM-Modal* [13], индустријски роботи вертикалне зглобне, односно антропоморфне конфигурације су данас најзаступљенији у обради резањем. Индустријске роботе, у поређењу са 5-осним машинама алаткама, одликује вишеструко нижа цена, велики радни простор и висока флексибилност. Међутим, изузетна комплексност програмирања задатака обраде, а поготову многоструко нижа крутост, односно тачност робота, у поређењу са машинама алаткама, представљају главне ограничавајуће факторе за ширу примену робота у обради. Из ових разлога, интензивна истраживања у области обраде роботима, која се данас спроводе у свету, односе се на решавање ове две групе проблема.

Сложеност програмирања робота, као и „нормално затворена“ архитектура комерцијално изведених управљачких система [14], представљају главни ограничавајући фактор за ширу примену робота у обради [15-18]. Роботски програмски језици различитих произвођача робота се значајно разликују по структури, кодној синтакси и семантици, као и

могућностима примене. У комбинацији са обучавањем су предвиђени за задатке код којих се једном написан програм користи дуже времена. Покривају врло различите области примене, као што су манипулација, бојење, заваривање, чишћење, полирање... Међутим, за примене робота у вишеосној обради, програмирање роботским програмским језицима је веома сложено и дуготрајно. Сложеност и разноврсност роботских програмских језика је последица недовољне кооперативности произвођача робота, у поређењу са произвођачима машина алатки који су прихвативши стандард RS274 (DIN 66025) поставили основ данашњих CNC система. Стандардизација у области програмирања робота је започета крајем 80-тих година прошлог века (*ICR - Intermediate Code for Robots* и *IRDATA - Industrial Robot Data*). Међутим, произвођачи робота и даље развијају своје програмске језике са одређеним специфичностима, нарочито за одређене области примене.

Данас су за програмирање робота у вишеосној обради комерцијално доступна различита софтверска решења, као што су транслатори програма обраде за машине алатке у роботске програмске језике, *CAM* постпроцесори за генерисање роботских програма и сл., која пружају могућност индиректног програмирања робота за обраду. Разноврсни софтверски пакети на тржишту, који омогућавају примену робота у обради, не нуде суштинско решење проблема, односно могућност да програмирање робота за обраду буде директно применљиво од стране програмера и оператера са искуством у CNC технологијама и програмирању у G-кôду. Као што је поменуто, данашње примене робота у вишеосној обради су базиране на комерцијалним управљачким системима и програмским језицима, намењеним за програмирање традиционалних роботских задатака [5-9, 17, 19] и/или техника обучавања робота[20-23], чија је примена ограничена углавном на реверзно инжењерство.

Вишегодишња истраживања у области развоја реконфигурабилних обрадних система на бази робота на Катедри за производно машинство Машинског Факултета Универзитета у Београду и Лола институту, у којима је учествовао и аутор, су обухватила решавање две групе проблема:

- Развој реконфигурабилног система управљања и програмирања

робота у G-кôду [24-26];

- Повећање тачности обраде компензацијом грешака насталих услед статичке попустљивости робота при обради [27, 28].

Полазну основу за анализу и решавање поменутих проблема је представљала поставка концепта реконфигурабилне обрадне ћелије на бази стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе и обрадном јединицом (моторним вретеном). Усвојени концепт реконфигурабилне обрадне ћелије је подразумевао генерализацију кинематичког моделирања, као основе за развој реконфигурабилног управљачког система са могућношћу програмирања обраде у G-кôду, што је основни циљ истраживања у овој дисертацији. Предложени приступ, поред могућности директног коришћења постојећих CAD/CAM система за програмирање 5-осних машина алатки, имао је за циљ и развој алгоритама и софтверских система за компензацију грешака у обради, које су последица статичке попустљивости робота [27, 28].

Управљање отворене архитектуре (OAC - *Open Architecture Control*) је широко прихваћен приступ у области управљања машинама алаткама, а у последње време и роботима. Данашњи комерцијални управљачки системи, декларисани као системи отворене архитектуре, базирани су на специфичним и недоступним управљачким алгоритмима и заштићеном софтверу произвођача. По правилу, ови системи су предимензионисани у погледу функција које нуде, како би покрили што шири спектар машина алатки или робота на тржишту. За конфигурисање и реконфигурисање комерцијалних система управљања, систем интеграторима су неопходна одговарајућа експертска знања, која су специфична за сваког конкретног произвођача. Из ових разлога се данас, као императив у развоју реконфигурабилних обрадних система, намеће потреба за управљачким системима отворене архитектуре, који могу бити реконфигурисани од стране корисника, како у софтверском, тако и у хардверском смислу. Могућност конфигурисања и реконфигурисања управљачког система зависи од отворености, модуларности и скалабилности управљачке архитектуре [29], која је у општем случају хијерархијски организована, табела 1.1.

Табела 1.1 Хијерархијски нивои софтвера управљачког система

I ниво	Интерпретери управљачких програма (<i>G</i> -кôд, <i>L-ILR</i> , <i>STEP-NC</i>)	
II ниво	Компоненте интерполатора	Секвенцијални управљачки модули
III ниво	Кинематички модули	
VI ниво	Софтверски модули спрежног хардвера	

У овој дисертацији су приказана два метода развоја реконфигурабилних управљачких система роботских ћелија за обраду:

- Метод развоја на бази готових хардверских и софтверских система отворене архитектуре;
- Метод развоја на бази софтверских алата и технологија отворене архитектуре и сопствених алгоритама управљања.

Оба метода се заснивају на *PC* хардверској архитектури и као саставни део система нуде могућност конфигурисања виртуелних роботских ћелија у циљу верификације, како програма обраде, тако и конфигурисаних роботских ћелија за дату обраду.

За истраживања, чији је део резултата приказан у овој дисертацији, полазну основу су чинили: (i) Научно-истраживачки и развојни пројекти Катедре за производно машинство, Машинског факултета у Београду [30-33] и Лола института [34-38], који су финансирани од стране Министарства просвете, науке и технолошког развоја, као и оквирних програма Европске уније (*FP7*), (ii) Сарадња са истраживачким центрима у земљи и иностранству, која је допринела актуелности приказаних истраживања и резултата, (iii) Дугогодишња традиција Лола института у развоју управљачких система за машине алатке и индустријске роботе; (iv) База знања и традиција у области индустријске роботике успостављена на Катедри за производно машинство, Машинског факултета у Београду; (v) Распоживи рачунарски хардвер, софтверске библиотеке и алати отворене архитектуре.

Истраживања у оквиру ове дисертације се заснивају на следећим хипотезама:

- Да је постојећим приступима пројектовања и организовања модуларних и реконфигурабилних роботских ћелија за обраду могуће конфигурисати различите структуре обрадних система на бази робота, комбинујући кинематичке моделе серијских робота

вертикалне зглобне конфигурације са додатним обртним и трансляторним осама за изабране класе делова;

- Да је могуће генерализовано кинематичко моделирање оваквих реконфигурабилних роботских ћелија за обраду за случајеве 3-осне, 4-осне, (3+2)-осне и 5-осне обраде глодањем, по конвенцијама за машине алатке, у циљу програмирања применом *G*-кôда, по стандарду *EIA (Electronic Industries Alliance) RS-274D*, генерисаног савременим *CAD/CAM* системима користећи и методе реверзног инжењерства;
- Да је избором типа и димензија вектора спољашњих координата, односно позиције и оријентације алата и/или радног предмета, могуће конфигурисање роботских ћелија са потребним осама за захтеване случајеве обраде;
- Да је за специфичне потребе могуће укључити и додатне, редундантне осе за обраду делова већих димензија и/или избегавање сингуларитета и колизија при обради мање приступачних површина;
- Да је могуће формирати хијерархијску структуру управљачких модула развијених на бази постојећих *open-source, real-time* софтверских библиотека и дефинисати морфолошки метод за конфигурисање управљачког система према захтевима корисника.

Актуелност истраживања у области реконфигурабилних система управљања отворене архитектуре индустријских робота и машина алатки потврђује већи број истраживачких пројеката покренутих у САД-у, Јапану и ЕУ [24-26, 39-49].

Текст дисертације је, поред уводног дела, организован у пет главних поглавља, у којима су представљена обављена истраживања, уз закључак који садржи осврт на остварене резултате и у коме су најављени могући правци даљих истраживања.

У другом поглављу је представљено актуелно стање истраживања и развоја у области обраде роботима. Препознати су основни проблеми у примени робота за вишеосну обраду сложених делова, који се односе на комплексност програмирања и незадовољавајућу крутост, односно попустљивост структуре робота. Представљене су технике програмирања

робота за обраду које се ослањају на постојеће програмске језике и одговарајуће транслаторе програма обраде. Приказани су савремени роботи реномираних произвођача који се користе за обраду. Освртом на традиционалне технике програмирања робота за обраду и анализом актуелних пројеката у овој области, указано је на потребу и актуелност истраживања и развоја реконфигурабилних управљачких система отворене архитектуре роботских ћелија за обраду са програмирањем у G -кôду. Као основа за развој генерализованог приступа кинематичког моделирања, представљен је концепт реконфигурабилне ћелије за вишеосну обраду на бази робота вертикалне зглобне конфигурације, из кога је изведена структура елементарне обрадне ћелије на бази робота која укључује робот, обрадну јединицу (моторно вретено) са одговарајућим интерфејсима и радни сто. Елементарна обрадна ћелија је конфигурисана тако да емулира различите 5-осне вертикалне и хоризонталне машине алатке код којих се оријентација алата остварује његовим ротацијама.

Као основа развоју реконфигурабилног система управљања роботских ћелија за обраду са програмирањем у G -кôду, у трећем поглављу је детаљно показан приступ генерализованог кинематичког моделирања структура елементарне обрадне ћелије на бази робота. Полазећи од геометријских модела структура елементарне ћелије на бази робота са 5 и 6 степени слободе, које емулирају 5-осне машине алатке, поставком одговарајућих координатних система робота и координатних система емулираних машина алатки, створени су предуслови за дефинисање унутрашњих и спољашњих координата робота, односно емулираних 5-осних машина алатки. С обзиром да се у дисертацији разматра више конфигурација елементарних обрадних ћелија на бази робота, на одговарајући начин су представљени вектори унутрашњих координата за све разматране случајеве робота са 5 и 6 степени слободе, као и углови оријентација алата са припадајућим матрицама ротација за све разматране случајеве емулираних 5-осних машина алатки. Потом су дата решења директног и инверзног кинематичког проблема елементарних ћелија на бази робота као емулираних 5-осних машина алатки, за све разматране случајеве. На крају трећег поглавља је приказан метод

анализе радног простора елементарних ћелија на бази робота са 5 и 6 степени слободе, као емулираних 5-осних машина алатки. Као пример је дата анализа радног простора елементарне ћелије за обраду на бази робота која емулира 5-осну вертикалну глодалицу конфигурације (X, Y, Z, A, B) за случајеве 3-осне и 5-осне обраде.

У четвртном поглављу су представљена два метода развоја софтверски оријентисаних реконфигурабилних управљачки система роботских ћелија за обраду. Како се развој управљачких система у овој дисертацији базира на данас доступним хардверским и софтверским решењима отворене архитектуре, најпре је дата општа класификација управљачких система машина алатки и робота отворене архитектуре, који могу бити делимично или потпуно отворени, у хардверском и/или софтверском смислу.

Први постављени метод, који се базира на примени готових хардверских и софтверских решења, примењен је током развоја реконфигурабилног управљачког система потпуно отворене архитектуре. Развијени су софтверски кинематички модули, на основу претходно добијених генерализованих решења инверзне и директне кинематике, и дефинисане различите конфигурације управљачког система за све разматране случајеве елементарне обрадне ћелије на бази робота. Посебна пажња је посвећена развоју виртуелних обрадних ћелија, као есенцијалних алата за *off-line* симулацију програма обраде на датој конфигурацији обрадне ћелије.

Други метод развоја реконфигурабилног управљачког система, који подразумева примену софтверских алата и технологија отворене архитектуре уз развој алгоритама и софтверских функција управљања, приказан је у наставку четвртог поглавља. Полазећи од платформски независног развоја алгоритама и функција управљања, преко поставке платформски независне структуре реконфигурабилног управљачког система, постављене су основе за имплементацију софтверских компонената и конфигуравање управљачког система. Одабиром одговарајућих софтверских алата, односно софтверског окружења које дефинише оквире у којима се извршавају претходно развијене функције, уз одговарајуће апликативни интерфејсе за размену информација, створени су услови за

дефинисање управљачке конфигурације, односно могућности конфигурисања/реконфигурисања управљачког система. Применом расположивих софтверских технологија и алата отворене архитектуре, креиране су софтверске компоненте као носиоци претходно развијених управљачких функција са особинама: (i) портабилности - могућности извршавања на различитим хардверским платформама и оперативним системима, (ii) дистрибутивности - извршавање компонента једног софтверског система на више хардверских платформи, (iii) конфигурабилности - могућности промене конфигурације компоненте пре и у току извршавања, (iv) реконфигурабилности - могућности укључивања/искључивања компоненте у систем када је то захтевано. Уз развијени подсистем за *off-line* програмирање и виртуелну симулацију, на бази савремених графичких софтверских библиотека, у овом поглављу је представљен део резултата, која у наредним фазама развоја треба да буду заокружена комплетним развојем G-код компилатора, имплементацијом STEP-NC стандарда, као и *on-line* метода компензације статичких грешака обраде, на бази интелигентних алгоритама модификације путање мерењем сила резања.

У циљу систематизације поступака иницијалног конфигурисања и измене управљачке конфигурације, како пре покретања, тако и током активног стања роботске ћелије, постављен је морфолошки метод за конфигурисање и реконфигурисање управљачког система. Применом одговарајућих дијаграма за моделирање динамичких процеса и успостављених релација између софтверских и хардверских компонента управљања, дефинисане су активности систем-интегратора, односно корисника, у процесу конфигурисања/реконфигурисања управљачког система.

Верификација експерименталног система управљања и програмирања реконфигурабилних обрадних ћелија на бази робота, развијеног применом првог метода, на бази готових хардверских и софтверских система отворене архитектуре, дата је у петом поглављу. Током извођења експеримената, потврђене су могућности реконфигурисања експерименталног управљачког

система обрадних ћелија на бази робота у циљу емулирања различитих 5-осних машина алатки. Експерименти су обухватили 3-осне и 5-осне обраде на различитим типовима емулираних 5-осних вертикалних и хоризонталних машина алатки одабраних типова радних предмета од мекших материјала, ниже и средње класе тачности: (i) тест радни предмети са аналитичким и слободним (енг. *freeform*) површинама слични радним предметима који се користе за испитивање 5-осних машина алатки, (ii) радни предмети са сложеним (аналитичким и *freeform*) функционалним површинама и (iii) радни предмети са сложеним естетским површинама. Већи број обрађених радних предмета је потврдио функционалност имплементираних кинематичких алгоритама, у одговарајућим конфигурацијама управљачког система обрадних ћелија на бази робота, чиме је верификован развијени систем управљања и програмирања на бази G-кода. Обрадом посебно пројектованог тест пробног дела, који захтева вишестрану обраду у једном базирању, демонстриране су могућности реконфигурисања роботске ћелије емулирањем различитих 5-осних машина алатки у току једног задатка, без физичких интервенција.

У поглављу шест је приказана експериментална верификација функционалности и перформанси реализованих основних софтверских компонената потребних за конфигурисање елементарне конфигурације управљачког система, применом постављеног другог метода, који је базиран на софтверским алатима и хардверским технологијама отворене архитектуре уз развој алгоритама управљања. Конфигурисани прототип је подвргнут тестирању функција и перформанси, најпре на одговарајућем тест лабораторијском систему са 3 аналогна серво погона, посебно развијеном за ове потребе. Затим је управљачки систем тестиран на реалном роботу одговарајућим тест програмима, извршавањем сложених програмираних вишеосних кретања. Тестирања су показала да је усвојени приступ развоју управљачког система, на бази софтверских алата и технологија отворене архитектуре уз развој алгоритама и функција управљања, дао очекиване резултате у погледу функција и перформанси развијеног експерименталног прототипа.

2 ВИШЕОСНА ОБРАДА РОБОТИМА

Потреба за технологијама вишеосне обраде глодањем великих делова од мекших материјала са сложеним естетским и функционалним површинама ниже и средње класе тачности је препозната у областима: израде делова од лаких легура, обраде дрвета и неметала (камен, пластика, стакло, композити), израде модела и калупа за језгра у ливарству, израде алата за производе од композита (корита чамаца, кабине возила, браника и сл.), рестаурације објеката културне баштине (цркве, манастири, споменици и сл.), као и израде позоришне и филмске сценографије. Чињеница да ове области покривају скупе 5-осне машине алатке је послужила као снажан подстицај истраживачима, произвођачима робота и *CAD/CAM* система да своје активности усмере у правцу истраживања и развоја обрадних система на бази робота, које одликује вишеструко нижа цена.

Темпо и правац развоја индустријских робота је највећим делом диктиран потребама аутомобилске индустрије у којој се примарни задаци за роботе односе на манипулацију и тачкасто заваривање. То је и разумљиво с обзиром на чињеницу да аутомобилска индустрија располаже највећим бројем робота у односу на остале индустријске гране. Према извештају *IFR-a (International Federation of Robotics)* [50], аутомобилска индустрија, континуирано повећава инвестиције у области примене робота, што потврђује све већи број инсталисаних робота, посебно у периоду 2010-2013. године. У свету се чине велики напори да се путем пројектног финансирања (нпр. *H2020*) и државним субвенцијама прошири примена робота и на друге области индустрије. Поред институционалних напора, технолошки напредак у области развоја робота, као и све нижа цена, учинили су роботе приступачним малим и средњим предузећима у различитим областима индустрије.

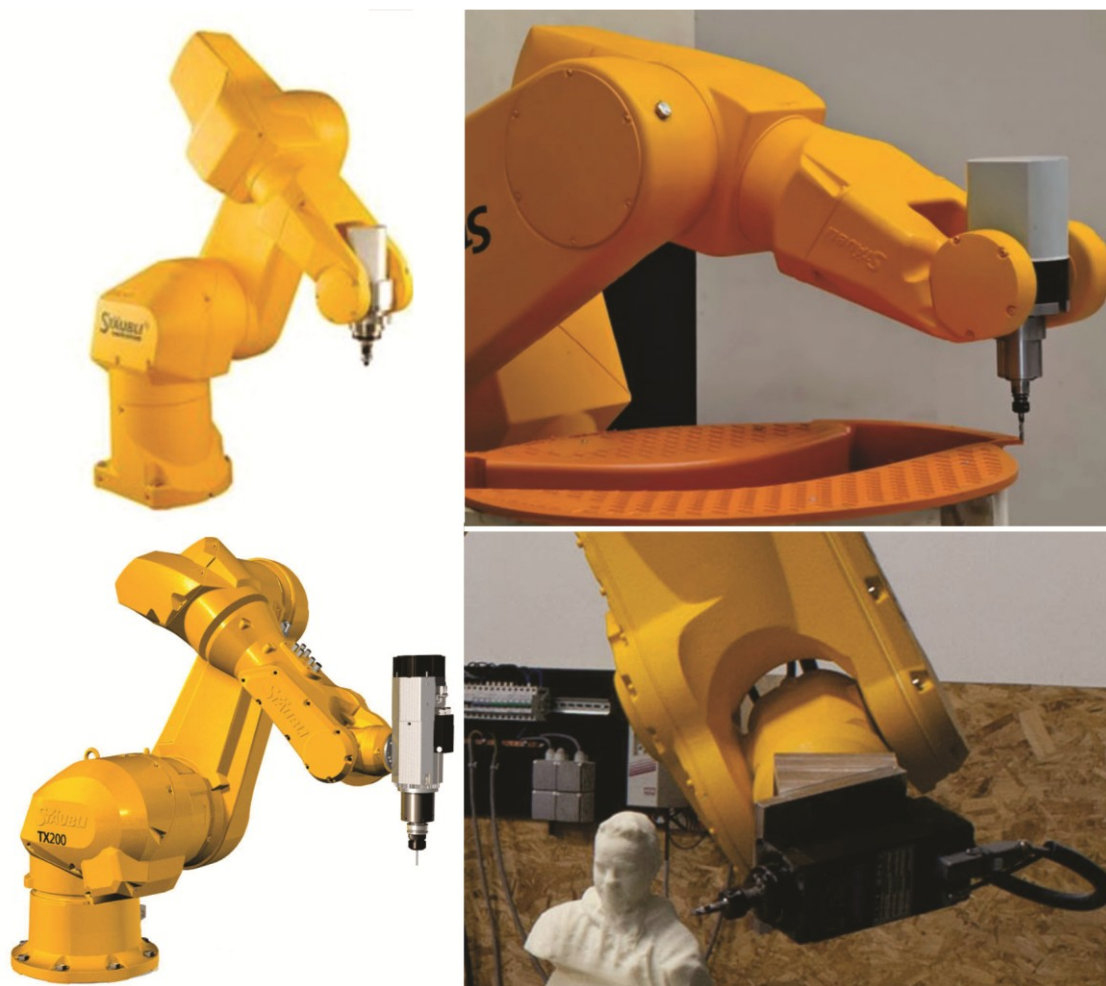
Индустријски роботи на данашњем нивоу развоја имају изузетне карактеристике у погледу флексибилности, носивости, тачности и динамике. С обзиром на чињеницу да су роботи по дефиницији мултифункционалне машине, управљачки системи и роботски програмски језици покривају врло

различите области примене робота, као што су манипулација, заваривање, бојење, тзв. помоћне операције обраде, као што су обарање ивица, чишћење, полирање и слично. Са становишта примене у обради резањем, роботе у поређењу са 5-осним машинама алаткама одликује већи радни простор и цена која је за ред величине нижа. Међутим, комплексност програмирања задатака обраде као и нижа крутост, а самим тим и тачност, представљају главне препреке за ширу примену робота у обради резањем. Као илустративни примери наведених проблема могу послужити наводи у [14] и [51]. Крутост машина алатки је, према [14], и до 50 пута већа од крутости индустријских робота зглобне конфигурације. За илустрацију сложености програмирања робота у [51] се наводи да је време потребно за брушење радног предмета 400 пута мање од времена потребног за програмирање дате обраде. Из наведених разлога актуелна истраживања су усмерена на решавање поменуте две групе проблема у области обраде роботима.

Програмирање робота за вишеосну обраду, на традиционалан начин, уз коришћење софтверских алата за превођење *G*-кôда или *CL* (*Cutter Location*) датотека у роботске програмске језике или применом техника обучавања (*teach-in*), представља веома сложен и дуготрајан процес. Роботски програмски језици су предвиђени за примене у којима се једном написан програм користи дуже времена. За разлику од машина алатки код којих је усвајањем стандарда *EIA RS274*, односно *G*-кôда за програмирање обраде, постављен основ развоју *CNC* управљања и програмирања, покушаји стандардизације у области програмирања и управљања роботима нису имали ефеката. Према [51] и [52] различитости система за програмирање, у почетним фазама развоја робота су биле последица ограниченог тржишта, а касније и неспремности произвођача робота да свој управљачки софтвер прилагоде предложеним стандардима. Разноликост и сложеност програмирања робота се негативно одразила на њихову примену у обради. Реномирани произвођачи робота (*KUKA*, *Motoman*, *Fanuc*, *Staubli*) и *CAD/CAM* софтвера (*Delcam*, *Mastercam*) су у циљу савладавања баријера у вези са програмирањем робота за вишеосну обраду резањем, покренули више пројеката из ове области. Добијени резултати се у виду софтверских решења

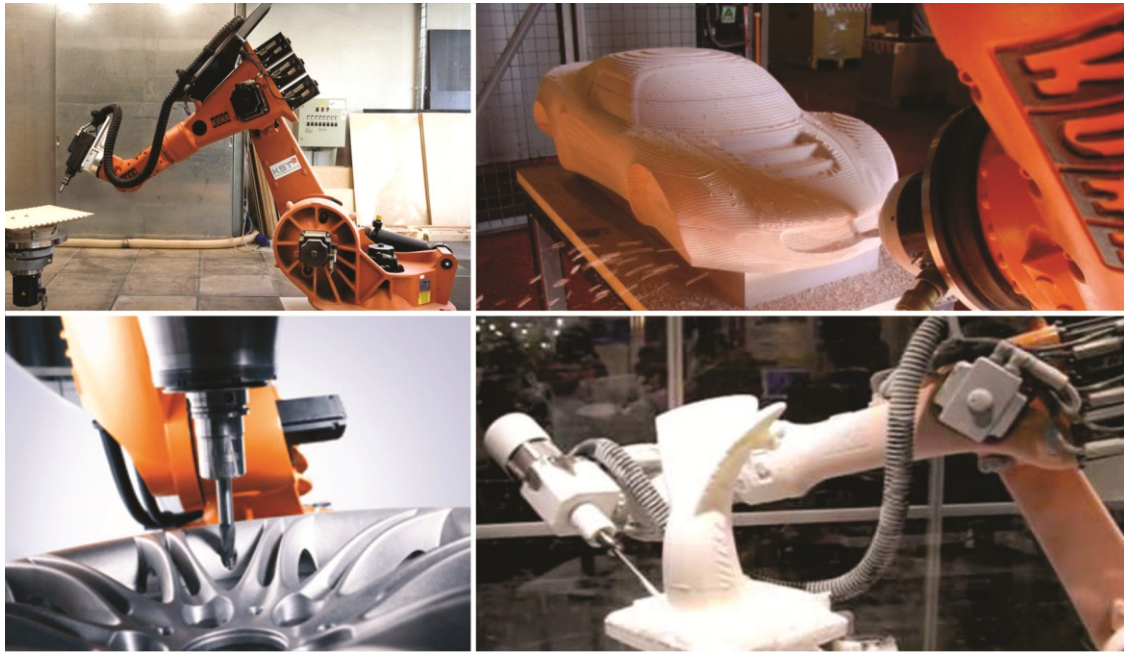
ВИШЕОСНА ОБРАДА РОБОТИМА

могу сврстати у две групе: (i) софтверски преводиоци програма обраде (G-кôда), генерисаних постојећим CAM системима за 5-осне машине алатке, на роботске програмске језике; (ii) поспроцесори CL датотека, који омогућавају директно генерисање програма обраде из стандардних CAM система на роботским програмским језицима. Комерцијализација поменутих резултата од стране произвођача CAM система (*Delcam, Mastercam,...*), донела је тржишту разноврсне постпроцесоре за генерисање програме обраде (*PowerMILL-Robot, Robomaster,...*) на програмским језицима различитих произвођача робота. Уз алате за виртуелну симулацију генерисаних роботских програма за обраду и софтверских система за оптимизацију путање алата, корисницима су комерцијално доступна комплетна решења за примену робота у обради. Примери комерцијалних робота за обраду резањем, реномираних произвођача, приказани су на сликама 2.1 до 2.4.

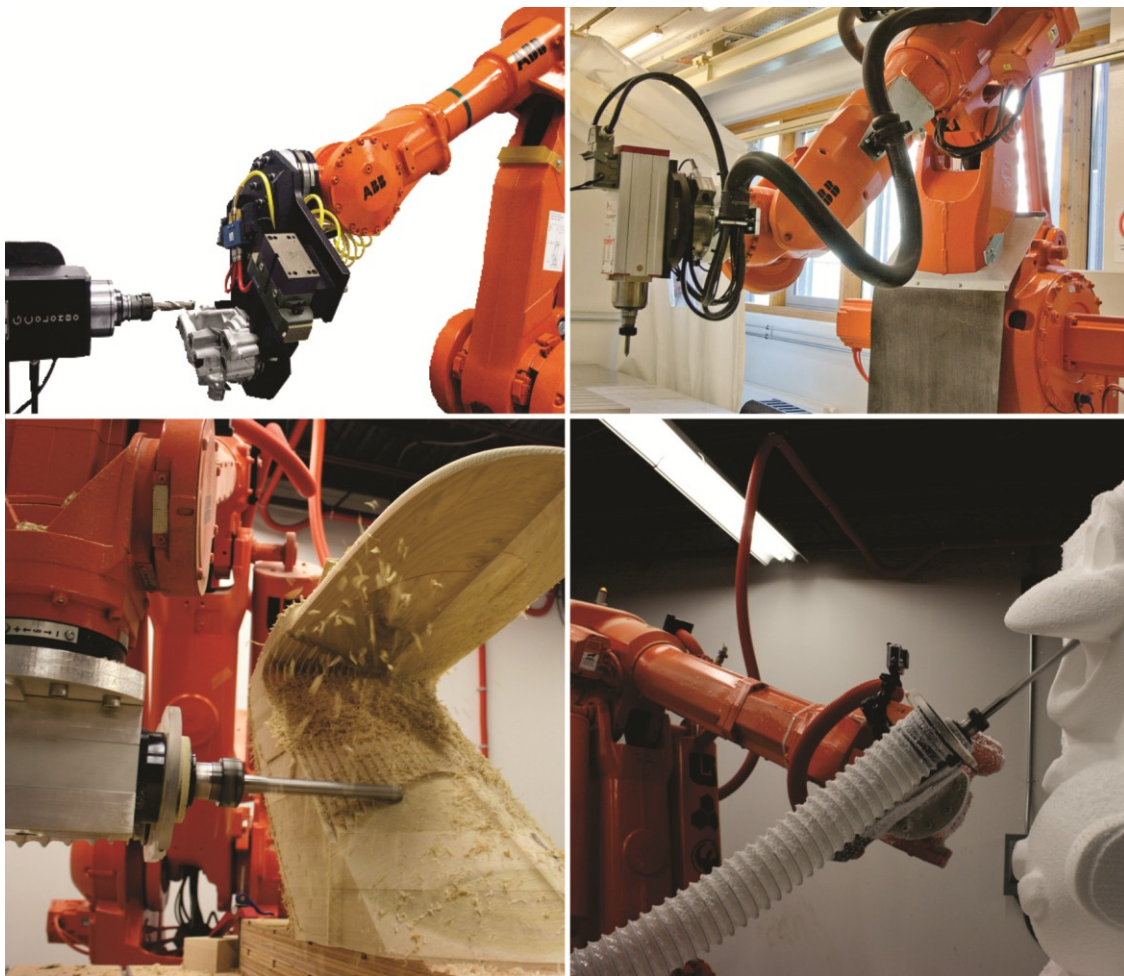


Слика 2.1 Примери робота за обраду произвођача *Staubli* [53]

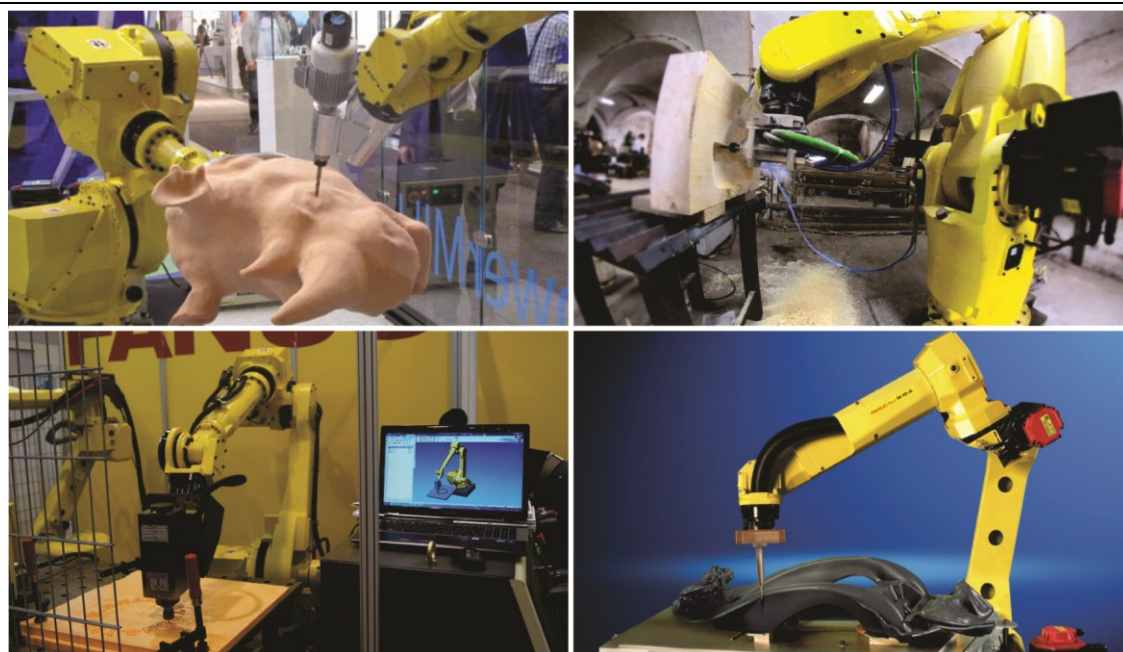
ВИШЕОСНА ОБРАДА РОБОТИМА



Слика 2.2 Примери робота за обраду произвођача *KUKA* [54]



Слика 2.3 Примери робота за обраду произвођача *ABB* [55]



Слика 2.4 Примери робота за обраду произвођача *FANUC* [56]

Поменуте технике превођења путање алата генерисане из *CAM*-а (из *G*-кôда или из *CL* датотеке) у роботске програмске језике умањују сложеност примене робота у вишеосној обради, али суштински не решавају проблем, односно могућност директне примене робота у обради од стране оператера и програмера који имају знања и искуства у *CNC* технологијама. Због тога је, како се наводи у [57], неопходно развити нове приступе у програмирању робота.

2.1 Управљање и програмирање робота за обраду

Традиционални приступи програмирању робота, обучавањем (*on-line*) и роботским програмским језицима (*off-line*), уз могућност симулације рада робота, поред примене за програмирање уобичајених роботских задатака, користе се и у техникама програмирања обраде роботима. Програмирање обраде обучавањем се обавља директно на роботу и од интереса је у поступцима реверзног инжењерства. Примена роботских програмских језика за програмирање обраде роботима се данас уобичајено базира на употреби *CAM* постпроцесора или одговарајућих транслятора *G*-кôда, уз могућност симулације кретања робота по програмираној путањи алата.

Током првих истраживања у вези са могућностима примене робота у

обради резањем [20-22], поступци за добијање програма обраде су се заснивали на коришћењу стандардних система обучавања робота (*teach-in*) [21]. Како је примена конвенционалног начина вођења робота у процесу програмирања обучавањем сложен и дуготрајан процес, поступци *on-line* програмирања робота за обраду су ограничени на мање сложене делове у реверзном инжењерству.

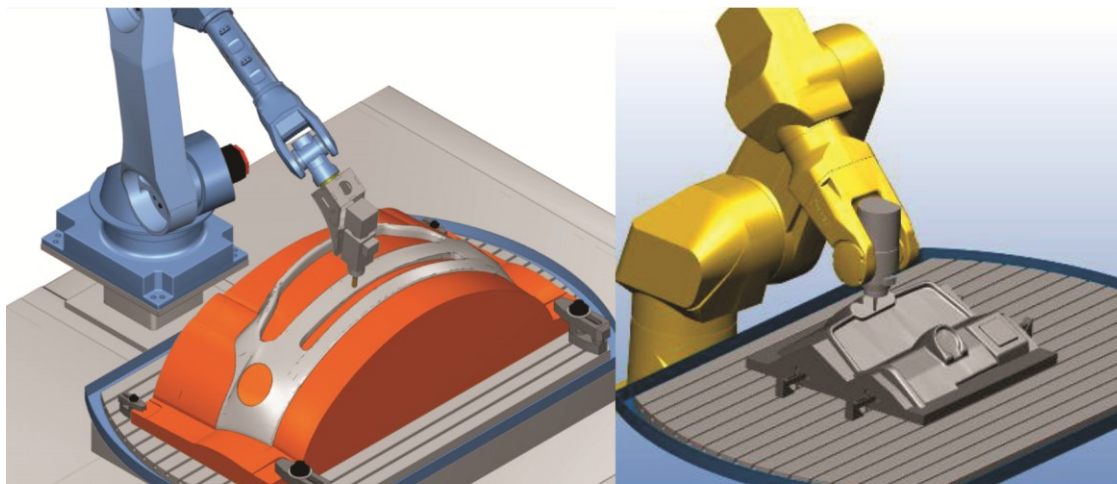
Примена специјализованих *CAM* система за генерисање програма обраде на програмским језицима робота, као и коришћење транслятора за превођење *G*-кôда у роботске програмске језике, представљају најзаступљеније технике програмирања робота за обраду. Савремена, комерцијално доступна софтверска решења комбинују *CAD/CAM* софтверске системе, постпроцесоре за превођење програма обраде (*G*-кôда или *CL* датотека) у роботски кôд и виртуелна окружења за симулацију и оптимизацију трајекторије алата у један софтверски систем, што значајно смањује време програмирања вишеосне обраде сложених делова. Интегрисана софтверска решења, која обједињују функције програмирања, симулације и генерисања роботског кôда донекле приближавају роботе за обраду *CNC* обрадним системима [58]. Данас је на тржишту присутно више комерцијалних софтверских пакета који омогућавају програмирање и примену робота у обради резањем, као што су:

- *KUKA.CAMRob* фирме *KUKA* [59];
- *Roboguide* фирме *FANUC* [60];
- *G-Code Converter EG* фирме *MOTOMAN* [61];
- *RobotWare Machining FC* фирме *ABB* [62];
- *ROBOTMASTER* фирме *Jabez Technologies* [63], слика 2.5а;
- *PowerMILL Robot* фирме *DELICAM* [64], слика 2.5б;

Поред комерцијално доступних решења, научно-истраживачке организације за своје специфичне потребе развијају сопствене системе управљања и програмирања, од којих се даље наводи неколико значајнијих.

У [19] је предложена роботска ћелија за обраду сложених просторних површина на припремцима великих димензија, слика 2.6, која укључује два робота вертикалне зглобне конфигурације са по 6 степени слободе.

Развијени систем за програмирање укључује постпроцесор који омогућава добијање програма обраде на роботском програмском језику из *CL* датотеке генерисане *Unigraphics CAM* системом.



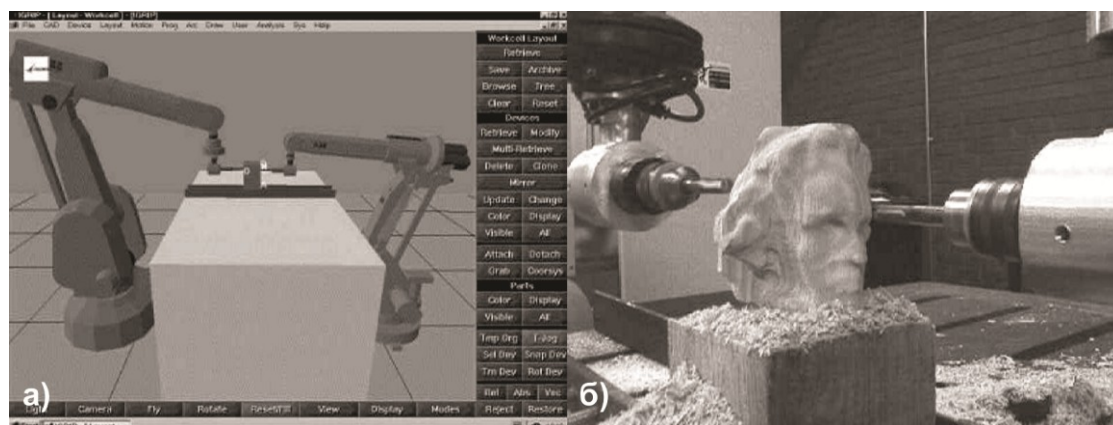
a) *Robomaster*



б) *PowerMILL Robot*

Слика 2.5 Примери комерцијалних софтверских система за програмирање и симулацију обраде роботима

Виртуелни симулатор програма обраде је интегрални део система за програмирање и омогућава да се у *off-line* режиму елиминишу колизије, слика 2.6а. Коришћење два робота у роботској ћелији омогућава знатно већу продуктивност и флексибилност у обради, као и поузданост система у случају отказа једног од робота.



Слика 2.6 Роботска ћелија за обраду са два робота

У радовима [65, 66] је представљена архитектура управљачког систем *RMAC (Reconfigurable Mechanisms for Application Control)* који има могућност управљања роботима у обради на основу директног интерпретирања *CL* датотеке као управљачког програма. *RMAC* је реализован као двопроесорска апликација. *CAD/CAM* софтвер са интегрисаним тзв. планером обраде се извршава на *Microsoft Windows* оперативном систему уз ангажовање једног од два процесорска језгра. Управљачки софтвер, који захтева строги временски детерминизам, извршава се на *real-time Linux* оперативном систему другог процесорског језгра. Комуникација између два система се остварује посредством дељених меморијских ресурса и позива системских функција апликативног програмског интерфејса. Интегрисање *CAD/CAM* и управљачког софтвера у једну хардверску платформу представља јединствен приступ у реализацији управљачког система са циљем да се изоставе кораци постпроцесирања и директног генерисања програма обраде за робота. Означен као реконфигурабилан, *RMAC* систем поседује могућност избора управљачке конфигурације посредством интегрисаног конфигурационог интерфејса.

Пројекат *COMET (COmponents and METhods – компоненте и методи за адаптивно управљање индустријским роботима у обради)* [67], разматра проблем недовољне крутости, као један од кључних недостатака робота у обради, кроз развој управљачког система на модуларном софтверском концепту, уводећи неколико важних иновација базираних на:

- Методологији моделирања кинематике и динамике индустријских робота у циљу предвиђања статичког и динамичког понашања механизма;
- Интегрисаном програмском и симулационом окружењу за генерисање програма обраде на роботским програмским језицима;
- Адаптивном оптичком праћењу система у циљу одређивања разлике између програмиране и остварене путање током обраде и корекције грешке у реалном времену;
- Алгоритму за динамичку компензацију (*High Dynamic Compensation Mechanism - HDCM*) који треба да омогући тачност већу од $50\mu\text{m}$, што је значајно боље од резултата које нуди сама структура робота.

Као основа за развој система за генерисање програма обраде на роботским програмским језицима, конзорцијуму *COMET* је послужио *PowerMILL Robot* интерфејс, производ фирме *Delcam*, која је и сама чланица конзорцијума и координатор пројекта.

Резултати пројекта *COMET* би требало да остваре уштеде корисницима од око 30% при обради роботима, уместо конвенционалним машинама алаткама. Планирано је да по завршетку пројекта, који је започет 2011. године, развијени систем буде интегрисан у управљачке системе познатих произвођача робота (*ABB, KUKA, Motoman*, итд.).

Анализом резултата, остварених постојећим техникама програмирања робота за обраду, може се закључити да:

- Приступи програмирању робота за обраду се највећим делом ослањају на постојеће софтверске системе и технике програмирања робота, па као такви захтевају оператере обучене за руковање роботима, као и познавање кодне синтаксе роботског програмског језика у случају потребе за изменама у програму непосредно пре и у току обраде;
- Хардвер савремених управљачких система индустријских робота је заснован на *PC* архитектури, било да се ради о комерцијалним системима или системима који су настали као плод научноистраживачког рада у академским институцијама. Универзална *PC* хардверска платформа даје могућност интегрисања различитих

сензора у управљачки систем, као и њихову фузију. Лака надоградња и промена хардверске конфигурације је једна од најважнијих особина отворених хардверских система;

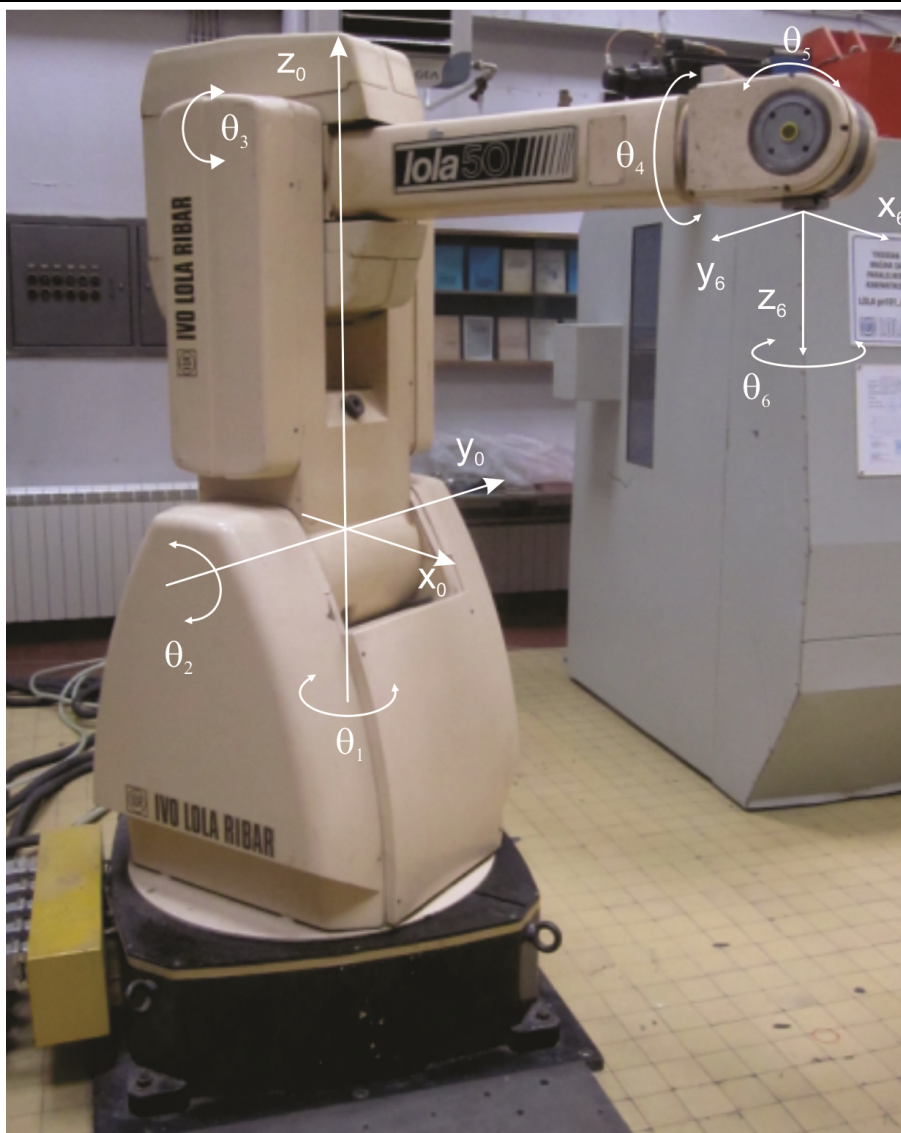
- Развој управљачког софтвера се највећим делом ослања на библиотеке и готова решења отворене архитектуре. Овакав приступ омогућава прецизну предикцију перформанси система и потребног времена за имплементацију управљања, према конкретним захтевима корисника;
- Највећи број управљачких система индустријских робота се у одређеном сегменту ослања на *Microsoft Windows* оперативни систем. Добра пракса подразумева да се *Windows* користи као основ за развој и конфигурисање графичког корисничког интерфејса, с обзиром на чињеницу да нуди препознатљиве графичке елементе. Покушаји да се реализују функције које *Microsoft Windows*-у дају временски детерминистичке одлике приликом извршавања апликација управљачког система, могу дати извесне резултате само у академским применама;
- Коришћење индустријских робота у обради резањем постаје актуелан начин да се класична и скупа машина алатка замени у сегменту обраде лаких материјала, као што су дрво, алуминијум, пластика. Нова истраживања ће показати [67] да индустријски робот у поменутом сегменту може имати задовољавајућу тачност, али са далеко већим радним простором и брзинама обраде у односу на конвенционалне машине алатке.

2.2 Поставка концепта реконфигурабилне ћелије за вишеосну обраду на бази робота вертикалне зглобне конфигурације

Идеја да се постојећи *CAD/CAM* системи користе за програмирање робота у *G*-кôду за вишеосну обраду резањем, наводи на приступ развоју управљачког система који треба да омогући емулацију хоризонталних и вертикалних 5-осних машина алатки, роботима вертикалне зглобне конфигурације. Структура реконфигурабилне роботске ћелије за обраду, која емулира хоризонталне и/или вертикалне 5-осне машине алатке, чини основу

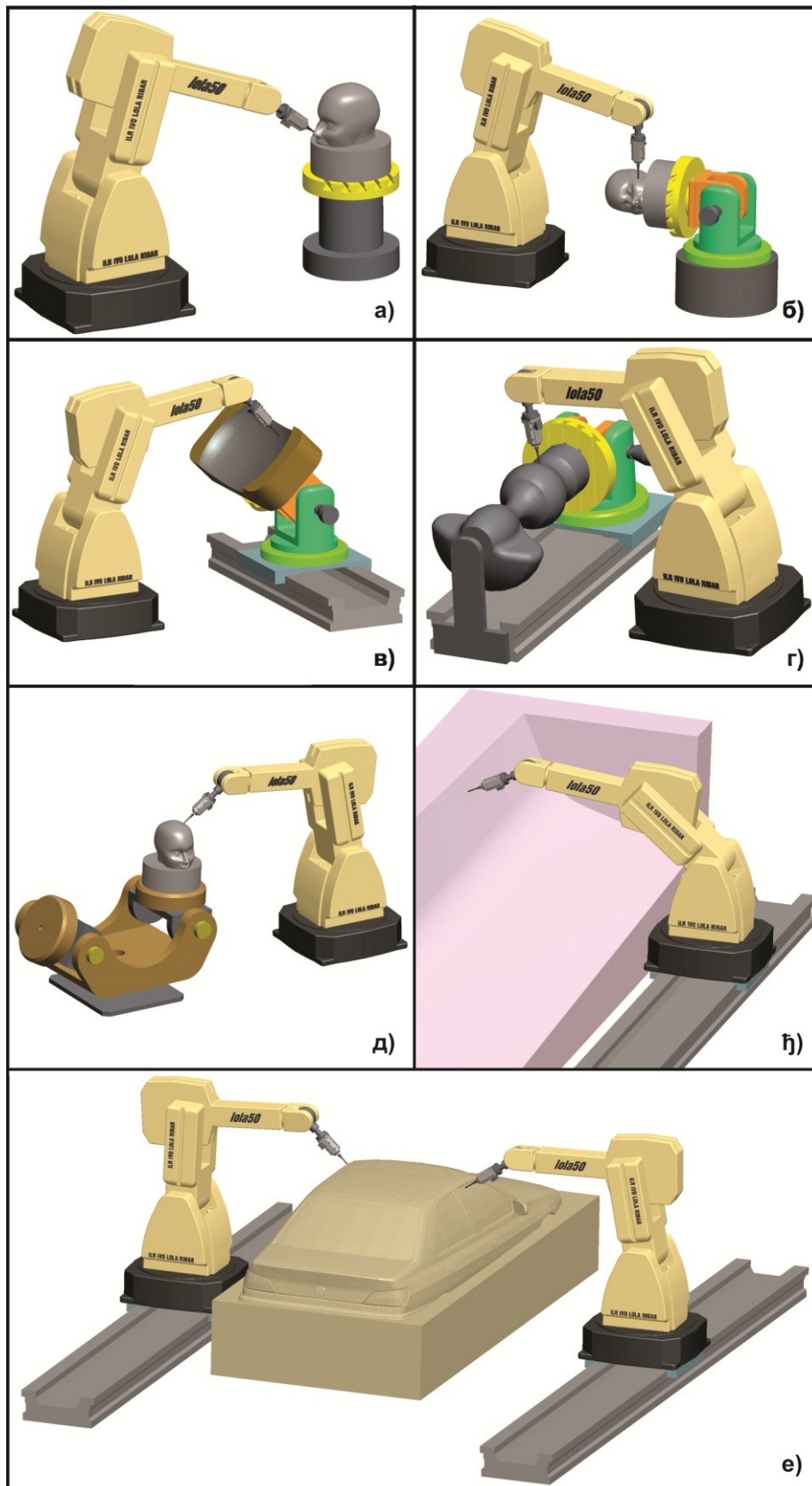
за развој софтверских и хардверских модула реконфигурабилног управљачког система. Типови 3-осних или 5-осних машина алатки које емулирају работи, као и број и распоред додатних оса директно дефинишу структуру софтверских кинематичких модула роботске ћелије, као и њихове везе са модулима суседних хијерархијских нивоа управљачког система. Овакав приступ подразумева идентификацију, класификацију и кинематичку анализу скупа изабраних конфигурација ћелије, која поред основне кинематичке структуре робота вертикалне зглобне конфигурације са 5 или 6 степени слободе, укључује и модуле додатних обртних и/или транслаторних оса. У циљу повећања радног простора, модул транслаторне осе може бити придодат основи робота, док се на појединачним модулима обртних и транслаторних оса, или њиховој комбинацији са одговарајућом кинематичком везом, поставља радни сто у циљу лакше обраде неприступачних делова радног предмета.

Комерцијално изведене роботске ћелије за обраду се базирају на комбинацији одговарајућих додатних, обртних и/или транслаторних оса и робота вертикалне зглобне конфигурације са 5 и 6 степени слободе. За поставку и анализу концепта реконфигурабилне ћелије за вишеосну обраду на бази робота вертикалне зглобне конфигурације са 5 и 6 степени слободе у комбинацији са додатним транслаторним и/или обртним осама, у овој дисертацији се разматра неколико типских структура ћелије, при чему работи емулирају 5-осне машине алатке како би се за програмирање обраде могао користити расположиви *CAD/CAM* систем. Експериментална провера и анализа постављеног концепта реконфигурабилне роботске ћелије за обраду се базира на расположивом експерименталном стандардизованом роботу вертикалне зглобне конфигурације домаће производње са 6 степени слободе, слика 2.7, који у различитим конфигурацијама емулира постојеће стандардизоване и специјализоване роботе за обраду са 5 степени слободе. Овако постављен и проверен концепт на експерименталном роботу има за циљ да покаже његову општост и применљивост на постојеће стандардизоване и специјализоване роботе за обраду.



Слика 2.7 Експериментални стандардизовани робот вертикалне зглобне конфигурације са 6 степени слободе Лола 50

Постављене типске структуре ћелије на бази расположивог робота са 6 степени слободе, слика 2.8, омогућавају: (i) лакшу обраду неприступачних површина радног предмета, избегавање сингуларитета и ограничења кретања у зглобовима, као и смањење времена обраде у смислу мањег броја постављања (базирања) радног предмета, слике 2.8а до 2.8г; (ii) базирање наредног припрема у току обраде у циљу смањења припремног времена, слика 2.8д; (iii) обраду радних предмета већих димензија, слике 2.8ђ и 2.8е; (iv) укључивање два робота у циљу повећања продуктивности (производности) ћелије, слика 2.8е.



Слика 2.8 CAD модели неколико примера структура обрадних ћелија на бази расположивог стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе

Као што се може видети са слике 2.8, роботом вертикалне зглобне конфигурације са додатним модулима континуално управљаних транслаторних и/или обртних оса, формирају се различите конфигурације обрадне ћелије, код којих се обрада може вршити комбинацијом кретања свих или само неких оса робота. При конципирању система управљања роботским ћелијама за обраду, који је представљен у овој дисертацији, и који би својом реконфигурабилношћу могао да покрије структуре ћелија приказане на слици 2.8, полази се од елементарне конфигурације која укључује само један робот и фиксни радни сто, при чему у току обраде сва позициона и помоћна кретања изводи робот, слика 2.9.



Слика 2.9 Елементарна конфигурација обрадне ћелије на бази расположивог робота вертикалне зглобне конфигурације

2.2.1 Опис концепта

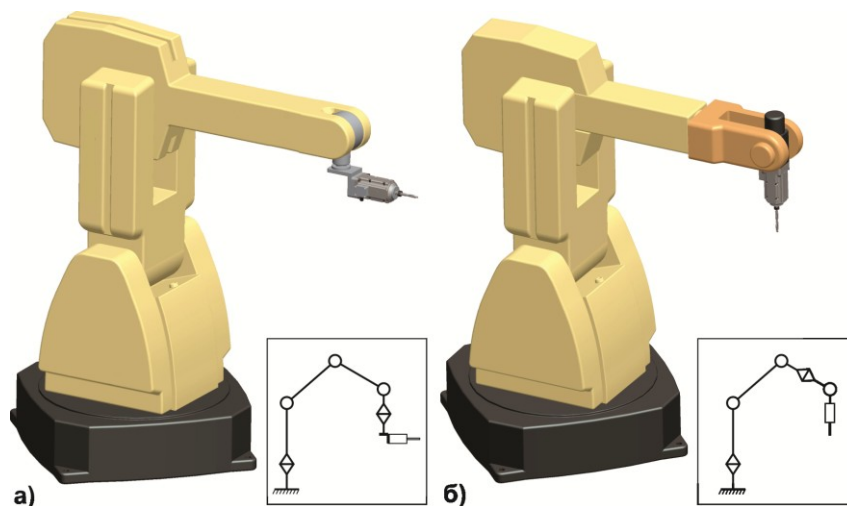
Постављени концепт реконфигурабилне роботске ћелије за вишеосну обраду већих делова, од мекших материјала, ниже и средње класе тачности у овој дисертацији је базиран на:

- Коришћењу стандардизованих робота вертикалне зглобне конфигурације са 5 степени слободе, слика 2.10а, специјализованих робота вертикалне зглобне конфигурације са 5 степени слободе и

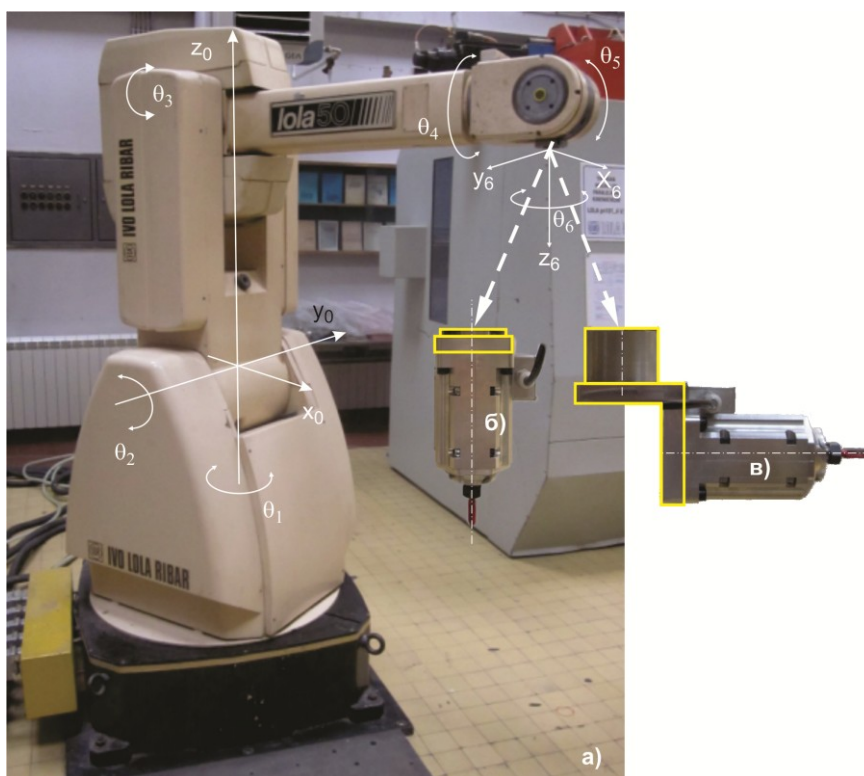
ВИШЕОСНА ОБРАДА РОБОТИМА

интегрисаним моторним вretenом, слика 2.10б, и стандардизованих робота вертикалне зглобне конфигурације са 6 степени слободе, слика 2.11;

- Коришћењу различито конфигурисаних додатни транслаторних и обртних оса за захтеване случајеве обраде, слика 2.8;



Слика 2.10 Примери стандардизованог и специјализованог робота за обраду вертикалне зглобне конфигурације са 5 степени слободе



Слика 2.11 Експериментални стандардизовани робот вертикалне зглобне конфигурације са додатним интерфејсима развијеним за потербе ових истраживања

- Коришћењу обрадне јединице (моторног вретена) са одговарајућим интерфејсима, слика 2.11;
- Развоју реконфигурабилног система управљања, са могућношћу програмирања обраде у G -кôду, који омогућава емулирање вертикалних и хоризонталних 5-осних машина алатки, код којих се оријентација алата остварује његовим ротацијама, уз коришћење постојећег *CAD/CAM* софтвера са стандардним постпроцесорима развијеним за дате емулиране машине алатке.

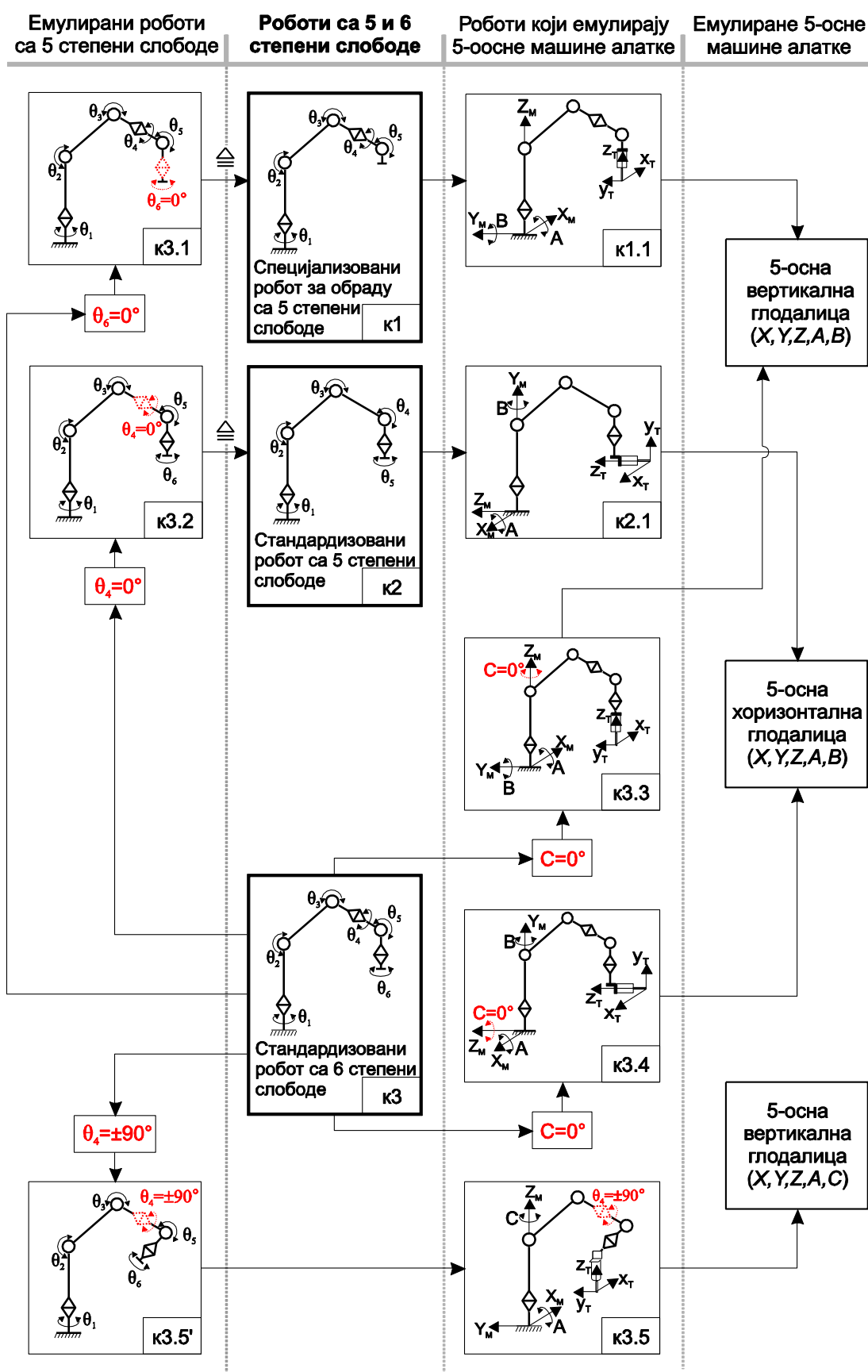
Конфигурисање елементарних ћелија на бази робота вертикалне зглобне конфигурације са 5 и 6 степени слободе, које емулирају различите типове хоризонталних и вертикалних машина алатки

Конфигурисањем елементарних обрадних ћелија, које укључују само један робот вертикалне зглобне конфигурације са 5 или 6 степени слободе и непокретни радни сто, могу се емулирати 5-осне хоризонталне и вертикалне машине алатке, код којих оријентацију алата остварују осе ротација главног вретена. Као што се види на слици 2.12, координатни системи овако емулираних 5-осних машина алатки одговарају стандардима [68, 69], чиме се омогућава директна примена постојећих *CAD/CAM* система за машине алатке.

На сликама 2.13 до 2.19 су показани примери емулираних 5-осних машина алатки роботима вертикалне зглобне конфигурације са 5 и 6 степени слободе, које са додатним обртним и/или транслаторним осама чине основу концепта реконфигурабилне ћелије за обраду у овој дисертацији.

За овако широко постављен концепт реконфигурабилне ћелије за обраду, развој реконфигурабилног система управљања и програмирања у овој дисертацији се базира на:

- Генерализованом приступу кинематичког моделирања реконфигурабилних роботских ћелија за обраду;
- Готовим софтверским и хардверским системима отворене архитектуре или софтверским технологијама и алатима отворене архитектуре и сопственим алгоритмима управљања;



Слика 2.12 Петоосне машине алатке емулиране роботима вертикалне зглобне конфигурације са 5 и 6 степени слободе

- Могућности програмирања обраде у G -кôду, односно коришћењу постојећих CAD/CAM система за машине алатке;
- Могућности конфигурисања еквивалентне виртуелне роботске ћелије у циљу финалне верификације програма обраде која укључује: (i) проверу позиције припремка унутар радног простора; (ii) проверу да ли робот, у одабраној конфигурацији, може да изврши програмирану трајекторију без прекорачења; (iii) визуелну детекцију колизија између сегмената робота, радног предмета, алата и помоћног прибора.

Могућност примене комерцијално расположивих робота вертикалне зглобне конфигурације са 5 и 6 степени слободе за емулацију 5-осних машина алатки у вишеосној обради, слика 2.12, са програмирањем у G -кôду применом стандардног CAD/CAM окружења, једна је од основних идеја ове дисертације. Даље се показују практични примери ових могућности, које ће бити обухваћене кинематичким моделирањем.

Комерцијални специјализовани робот за обраду са 5 степени слободе и интегрисаним моторним вретеном, конфигурација к1.1 на слици 2.12, емулира 5-осну вертикалну глодалицу конфигурације (X,Y,Z,A,B) , слика 2.13.

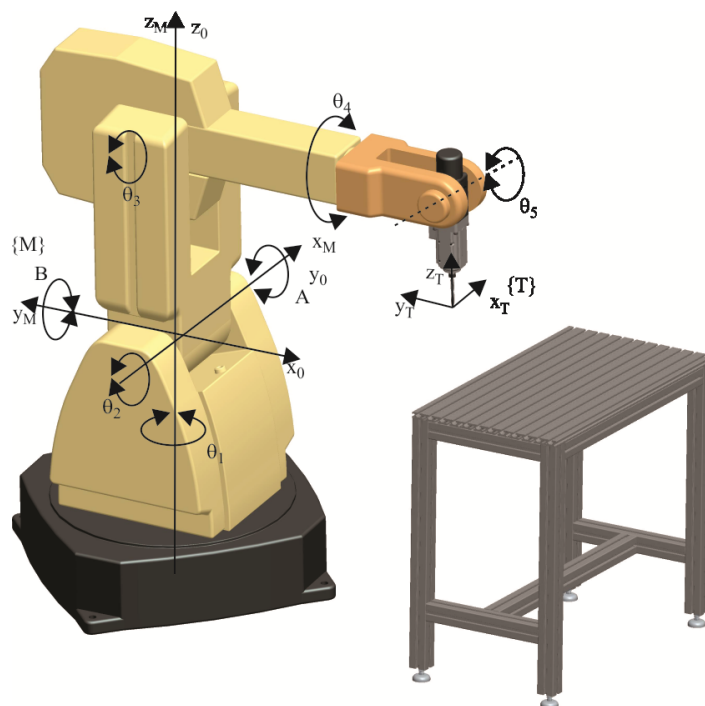
Комерцијални стандардизовани робот са 5 степени слободе, конфигурација к2.1 на слици 2.12, може емулирати хоризонталну глодалицу конфигурације (X,Y,Z,A,B) , слика 2.14.

Комерцијални стандардизовани роботи са 6 степени слободе, слика 2.12, могу емулирати 5-осне машине алатке следећих конфигурација:

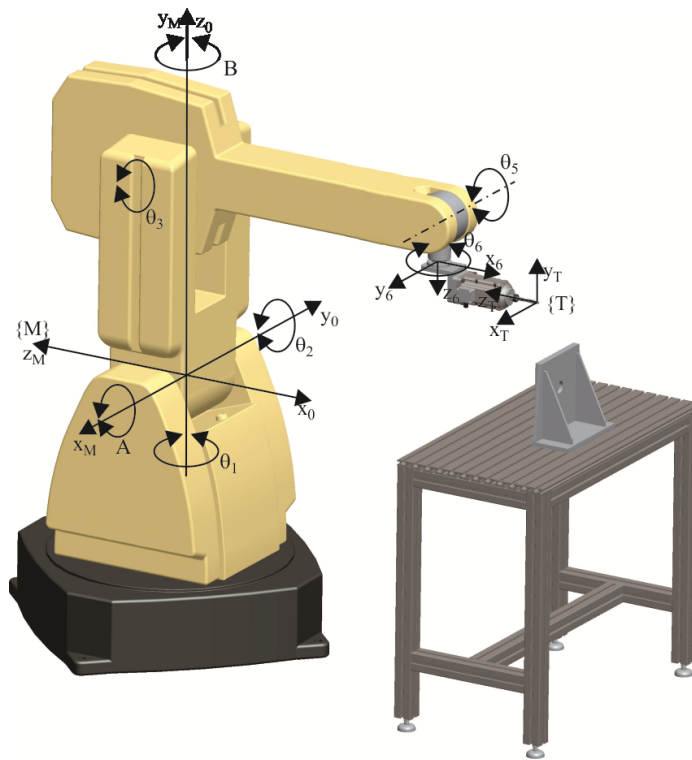
- 5-осне вертикалне глодалице (X,Y,Z,A,B) , при чему је оса робота θ_6 блокирана (нпр. $\theta_6 = 0$), конфигурација к3.1 на слици 2.15. Овде треба уочити да блокирањем осе θ_6 , робот са 6 степени слободе емулира специјализованог робота са 5 степени слободе, чиме се проблем своди на већ разматрану конфигурацију к1.1 са слике 2.13;
- 5-осне хоризонталне глодалице (X,Y,Z,A,B) , при чему је оса робота θ_4 блокирана (нпр. $\theta_4 = 0$), конфигурација к3.2 на слици 2.16. Овде треба уочити да блокирањем осе θ_4 , робот са 6 степени слободе емулира стандардизованог робота са 5 степени слободе, чиме се проблем своди на већ разматрану конфигурацију к2.1 са слици 2.14;

- 5-осне вертикалне глодалице (X,Y,Z,A,B) , при чему су све осе робота активне, а угао оријентације алата $C = const$, конфигурација к3.3 са слике 2.17.
- 5-осне хоризонталне глодалице (X,Y,Z,A,B) , при чему су све осе робота активне, а угао оријентације алата $C = const$, конфигурација к3.4 са слике 2.18;
- 5-осне вертикалне глодалице (X,Y,Z,A,C) , конфигурација к3.5 приказана на слици 2.19, при чему се оса робота θ_4 држи у константном положају $+90^\circ$ или -90° .

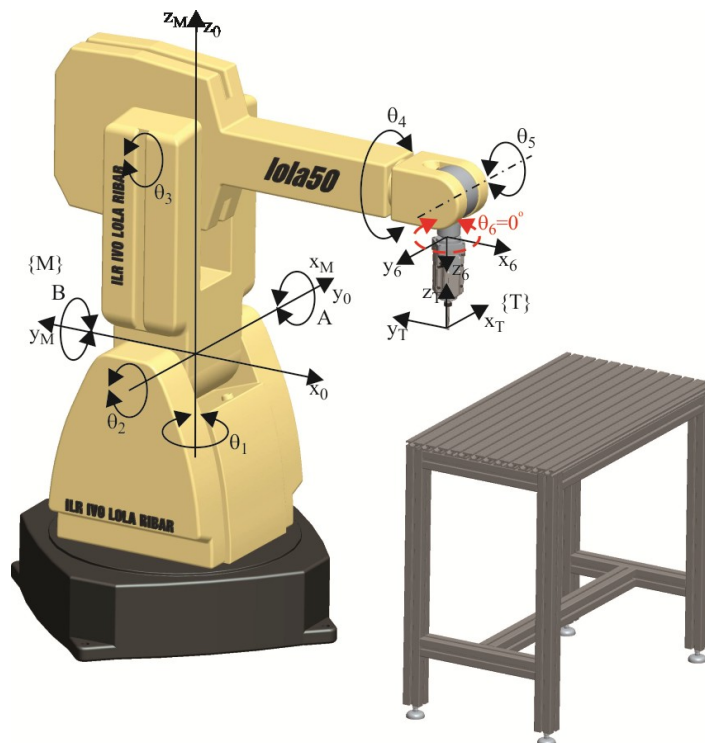
Развој генерализованог приступа кинематичког моделирања треба да омогући да се роботом са 6 степени слободе могу емулирати роботи са 5 и 6 степени слободе на начин који омогућава повећање опсега одређених углова оријентације алата, а у циљу емулације одговарајуће машине алатке. Слика 2.19 приказује конфигурацију к3.5 у којој је робот са 6 степени слободе разматран као 5-осна вертикална глодалица типа (X,Y,Z,A,C) са љуљањем главног вретена, при чему је оса θ_4 блокирана у позицији $+90^\circ$ или -90° .



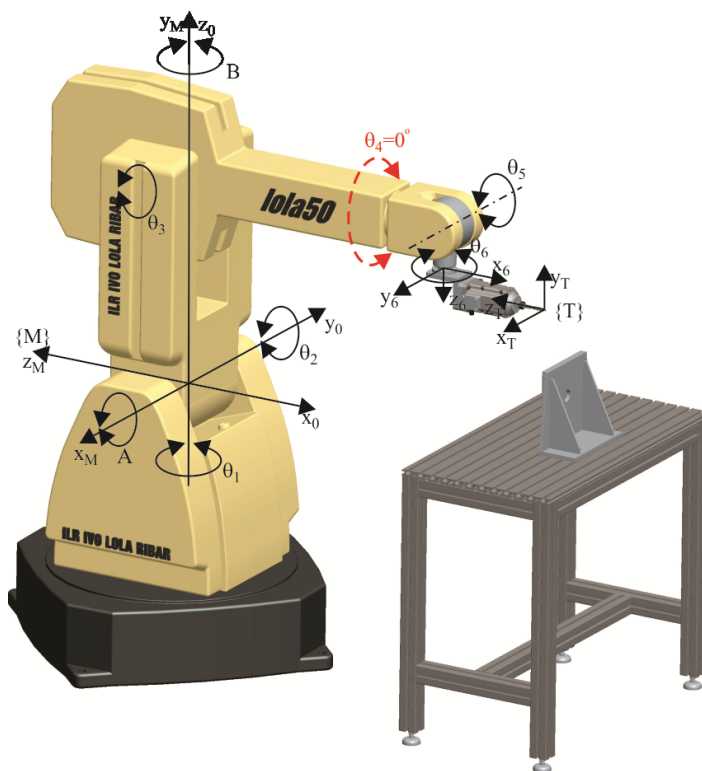
Слика 2.13 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) специјализованим роботом са 5 степени слободе (конфигурација к1.1)



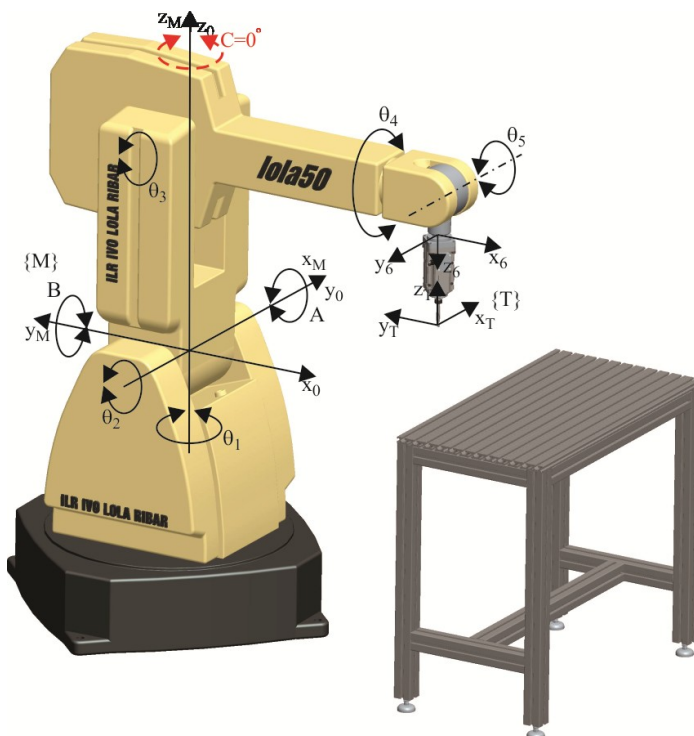
Слика 2.14 Емулирана 5-осна хоризонтална глодалица (X, Y, Z, A, B) стандардизованим роботом са 5 степени слободe (конфигурација к2.1)



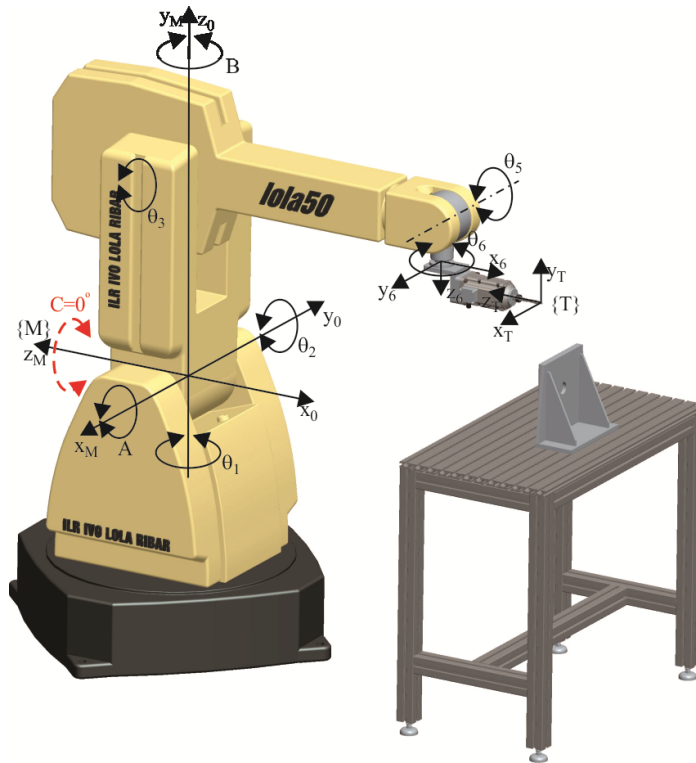
Слика 2.15 Емулирана 5-осна вертикална глодалица (X, Y, Z, A, B) стандардизованим роботом са 6 степени слободe, $\theta_6=0^\circ$ (конфигурација к3.1)



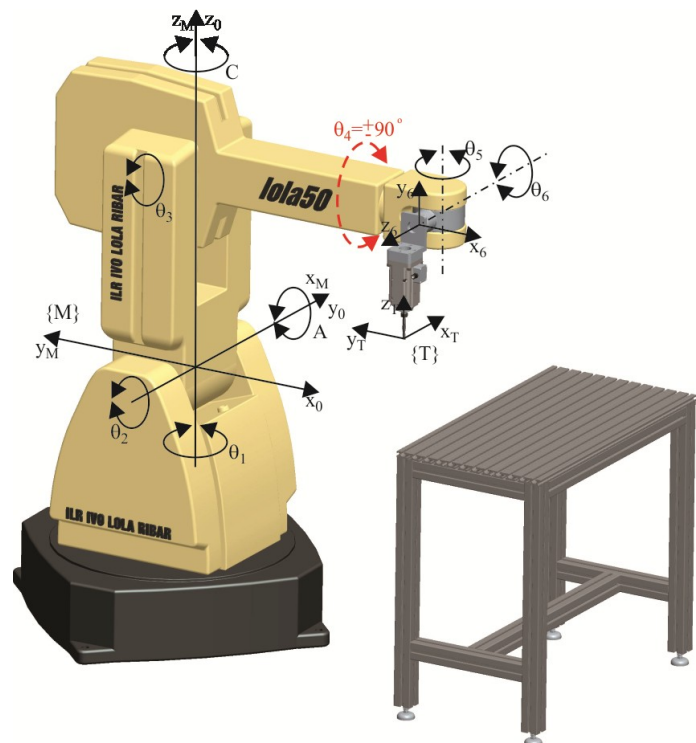
Слика 2.16 Емулирана 5-осна хоризонтална глодалица (X, Y, Z, A, B) стандардизованим роботом са 6 степени слободe, $\theta_4=0^\circ$ (конфигурација к3.2)



Слика 2.17 Емулирана 5-осна вертикална глодалица (X, Y, Z, A, B) стандардизованим роботом са 6 степени слободe, при чему су све осе робота су активне ($C=0^\circ$) (конфигурација к3.3)

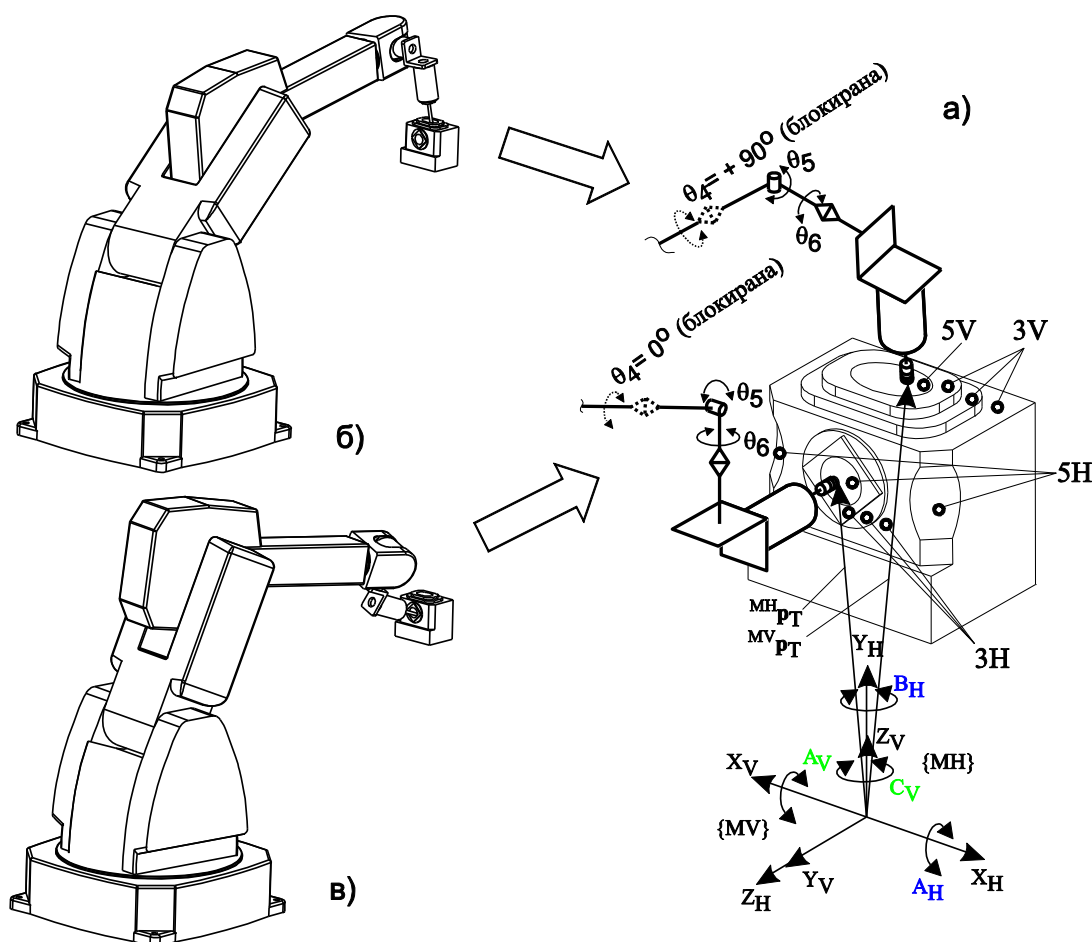


Слика 2.18 Емулирана 5-осна хоризонтална глодалица (X, Y, Z, A, B) стандардизованим роботом са 6 степени слободe, при чему су све осе робота су активне ($C=0^\circ$) (конфигурација к3.4)



Слика 2.19 Емулирана 5-осна вертикална глодалица (X, Y, Z, A, C) стандардизованим роботом са 6 степени слободe, $\theta_4 = \pm 90^\circ$ (конфигурација к3.5)

Као што је поменуто, предложени концепт реконфигурабилне обрадне ћелије на бази робота подразумева развој генерализованог приступа кинематичког моделирања, који треба да омогући реконфигурисање 6-осног робота, односно емулирање и хоризонталне и вертикалне 5-осне машине алатке. Ово подразумева могућност реконфигурисања и у току једног задатка, који захтева вишестрану обраду врло сложених делова у једном постављању, односно базирању (енг. *multifaceted machining*). На овај начин се избегавају проблеми сингуларитета робота и ограничених кретања у зглобовима. Као пример, на слици 2.20а је показан радни предмет код кога је потребно површине са горње стране обрадити програмима за 5-осну вертикалну глодалицу типа $(X_V, Y_V, Z_V, A_V, C_V)$, слика 2.20б, а сложене површине на бочним странама обрадити програмима за 5-осну хоризонталну глодалицу типа $(X_H, Y_H, Z_H, A_H, B_H)$, слика 2.20в.



Слика 2.20 Емулирање хоризонталне и вертикалне 5-осне машине алатке, стандардизованим роботом са 6 степени слободе, приликом обраде сложене површине са једним базирањем припремка

3 ГЕНЕРАЛИЗОВАНИ ПРИСТУП КИНЕМАТИЧКОГ МОДЕЛИРАЊА ЕЛЕМЕНТАРНЕ ЋЕЛИЈЕ ЗА ОБРАДУ НА БАЗИ РОБОТА ВЕРТИКАЛНЕ ЗГЛОБНЕ КОНФИГУРАЦИЈЕ

Током извршавања програма обраде написаног у G -кôду, управљачки систем робота за обраду интерполира програмирану путању у координатном систему емулиране машине алатке $\{M\}$, слике 2.13 до 2.19. У сваком интерполационом периоду се у реалном времену решава инверзни кинематички проблем (IKP) робота, при чему се прво позиција и оријентација алата из координатног система $\{M\}$ емулиране 5-осне машине алатке трансформише у спољашње координате робота, изражене у његовом базном координатном систему $\{0\}$, слике 3.1 до 3.7. Да би се могло управљати позицијом и оријентацијом алата у реалном времену, полази се од кинематичког модела робота за обраду, који укључује и кинематички модел 5-осне машине алатке која се емулира.

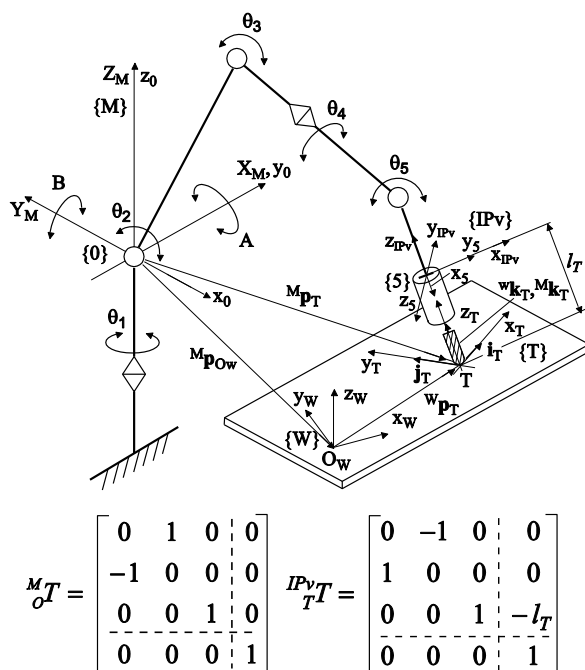
За генерализовани приступ кинематичког моделирања реконфигурабилне елементарне ћелије за обраду на бази робота вертикалне зглобне конфигурације са 5 и 6 степени слободе, слике 2.13 до 2.19, коришћен је расположиви стандардизовани робот вертикалне зглобне конфигурације са 6 степени слободе Лола 50, носивости 50 kg , слика 2.11. Робот је опремљен моторним вретеном са максималним бројем обртаја од 18000 min^{-1} , које је посредством одговарајућих додатних интерфејса повезано са стандардним интерфејсом робота, слика 2.11.

3.1 Геометријски модели структура елементарних ћелија на бази робота са 5 и 6 степени слободе, које емулирају 5-осне машине алатке

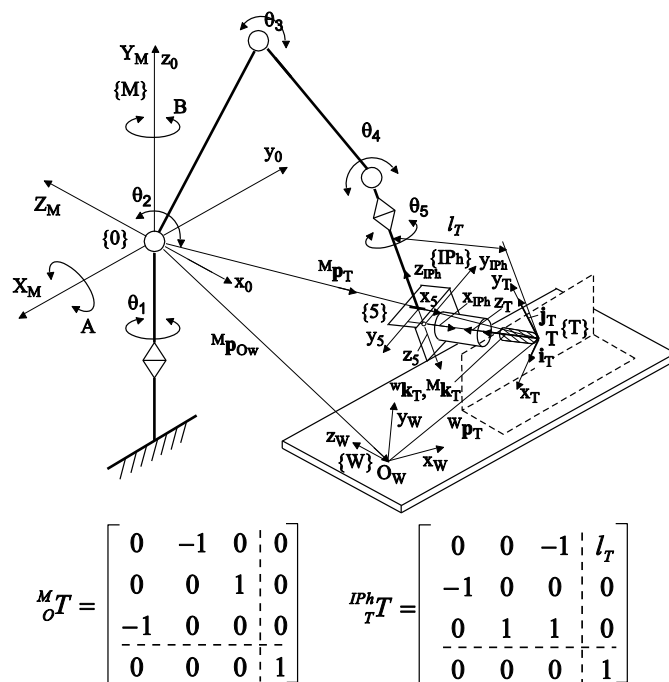
Геометријски модели структура елементарних ћелија на бази робота, односно емулираних 5-осних машина алатки, које су у претходном поглављу представљене на сликама 2.13 до 2.19, приказани су на сликама 3.1 до 3.7. На датим геометријским моделима су приказани следећи координатни системи:

- Базни координатни системи робота са 5 и 6 степени слободe {0};
- Координатни системи стандардних интерфејса робота (x_5, y_5, z_5) или (x_6, y_6, z_6) , означени као {5} или {6};
- Координатни системи додатних интерфејса $(x_{IP_v}, y_{IP_v}, z_{IP_v})$, $(x_{IP_h}, y_{IP_h}, z_{IP_h})$ и $(x_{IP_v'}, y_{IP_v'}, z_{IP_v'})$ за везу моторног вретена са стандардним интерфејсима робота, означени као {IP_v}, {IP_h} и {IP_{v'}}, уведени су да би углови оријентације интерфејса ψ , θ , φ у референтном положају робота, били $\psi = \theta = \varphi = 0^\circ$, што омогућава директну проверу једначина директне кинематике робота са 5 степени слободe који је изведен из кинематичког модела робота са 6 степени слободe;
- Координатни системи емулираних машина {M};
- Координатни системи алата {T};
- Координатни системи радних предмета {W}, који су усвојени произвољно на радном предмету.

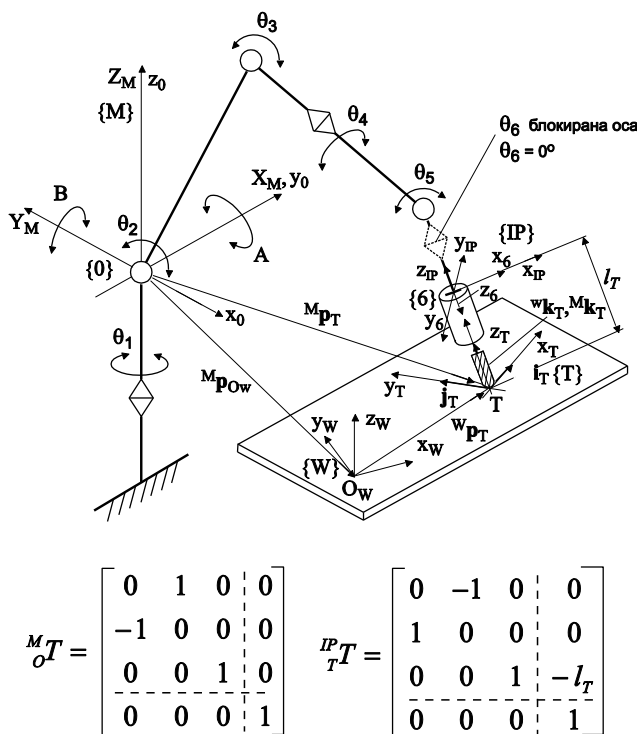
На сликама 3.1 до 3.7 су вектори \mathbf{v} у координатним системима {M} и {W} означени са ${}^M\mathbf{v}$ и ${}^W\mathbf{v}$, док су односи координатних система {0} и {M}, {T} и {IP} описани матрицама хомогених трансформација M_0T , ${}^{IP}_T T$, ${}^{IP_h}_T T$ и ${}^{IP_v}_T T$.



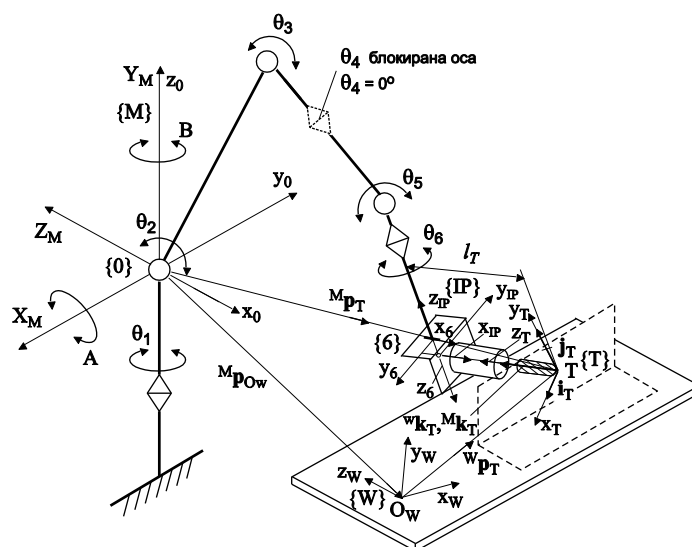
Слика 3.1 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) специјализованим роботом са 5 степени слободe (конфигурација к1.1)



Слика 3.2 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 5 степени слобде (конфигурација к2.1)

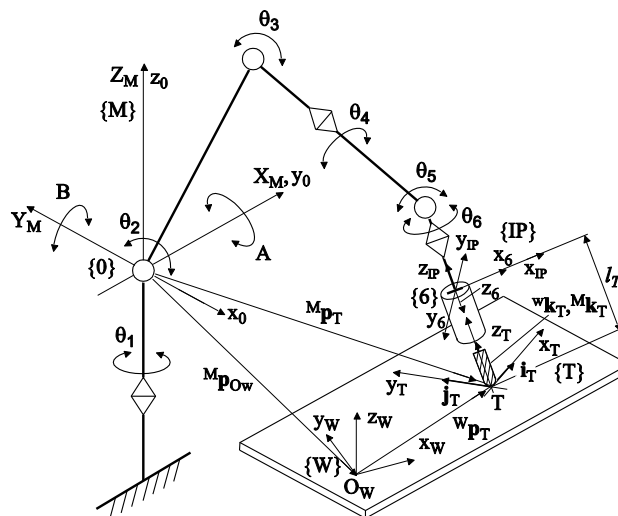


Слика 3.3 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слобде, $\theta_6=0^\circ$ (конфигурација к3.1)



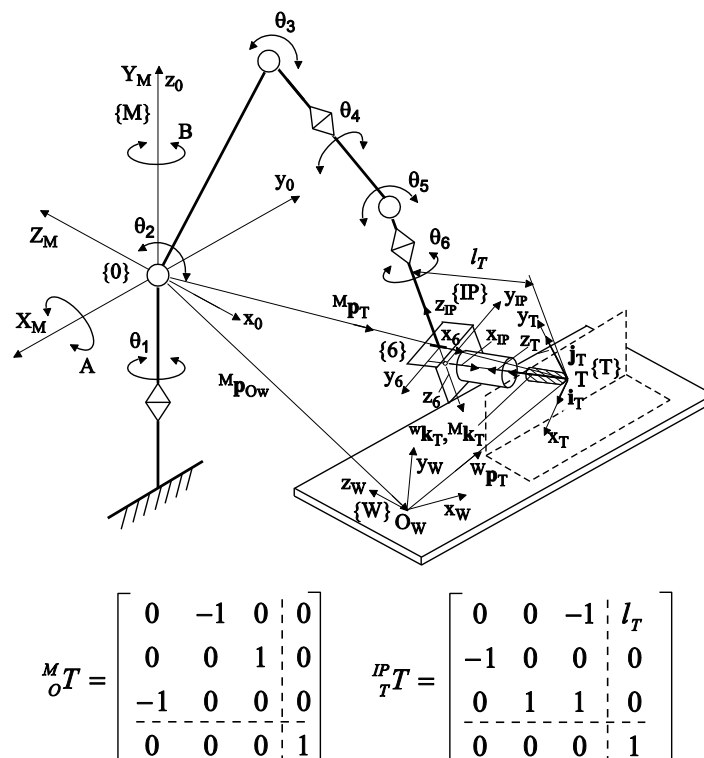
$${}^M_O T = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^I_P T = \begin{bmatrix} 0 & 0 & -1 & l_T \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

Слика 3.4 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободe, $\theta_4=0^\circ$ (конфигурација к3.2)

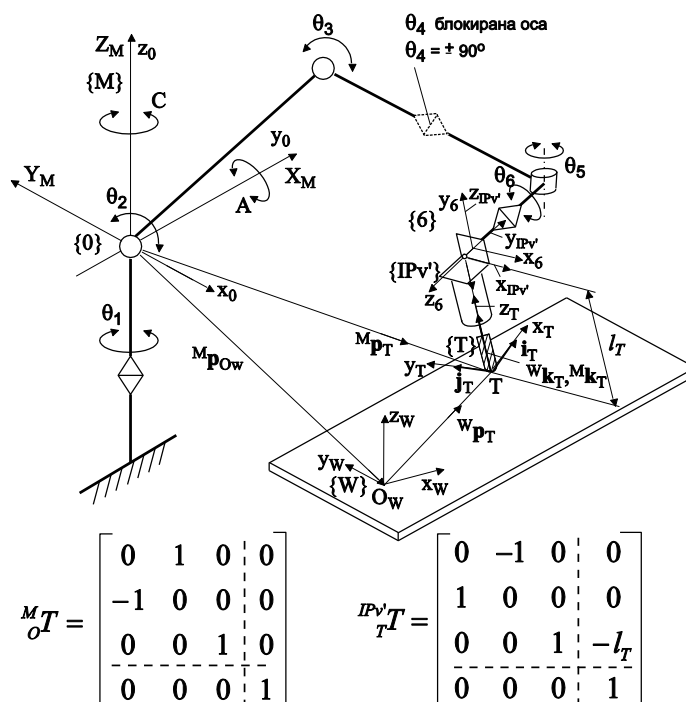


$${}^M_O T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^I_P T = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -l_T \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$

Слика 3.5 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободe, при чему су све осе робота су активне ($C=0^\circ$) (конфигурација к3.3)



Слика 3.6 Емулирана 5-осна хоризонтална глодалица (X,Y,Z,A,B) стандардизованим роботом са 6 степени слободe, при чему су све осе робота су активне (C=0°) (конфигурација к3.4)



Слика 3.7 Емулирана 5-осна вертикална глодалица (X,Y,Z,A,C) стандардизованим роботом са 6 степени слободe, θ₄=±90° (конфигурација к3.5)

Кинематичко моделирање стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе, на примеру робота Лола 50, обухвата решавање директног и инверзног кинематичког проблема за све конфигурације елементарне обрадне ћелије, односно емулираних машина алатки приказаних на сликама 2.13 до 2.19, односно на сликама 3.1 до 3.7. Расположиви робот вертикалне зглобне конфигурације је посматран као елементарна обрадна ћелија, односно као емулирана хоризонтална или вертикална 5-осна машина алатка и то у случајевима:

- у којима стандардизовани робот са 6 степени слободе емулира специјализоване ($\theta_6 = 0^\circ$) и стандардизоване ($\theta_4 = 0^\circ$) роботе са 5 степени слободе, слике 3.3 и 3.4,
- када стандардизовани робот са 6 степени слободе има све осе активне, при чему се одржава константним један од углова оријентације алата, слике 3.5 и 3.6,
- када стандардизовани робот са 6 степени слободе емулира специфичну конфигурацију робота са 5 степени слободе ($\theta_4 = \pm 90^\circ$), слика 3.7.

Приказани геометријски модели елементарних ћелија за обраду на бази робота, уз одговарајућа ограничења кретања оса зглобова, треба да обезбеде: (i) погодности везане за крутост робота, (ii) узимање у обзир само једног решења инверзног кинематичког проблема и (iii) избегавање сингуларних конфигурација робота унутар усвојеног радног простора [70].

3.2 Спољашње и унутрашње координате

За решавање директног и инверзног кинематичког проблема, најпре је потребно дефинисати унутрашње и спољашње координате робота, односно емулираних 5-осних машина алатки, слике 3.1 до 3.7.

3.2.1 Вектори унутрашњих координата

Вектори унутрашњих координата робота за обраду су представљени као:

- $\theta = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4 \ \theta_5]^T$ (3.1)

за специјализовани робот за обраду са 5 степени слободе који у

конфигурацији к1.1, слика 3.1, емулира вертикалну машину алатку, као и за стандардизованог робота са 5 степени слободe који у конфигурацији к2.1, слика 3.2, емулира хоризонталну машину алатку;

- $\theta = [\theta_1 \theta_2 \theta_3 \theta_4 \theta_5 \theta_6]^T$ (3.2)

за робот вертикалне зглобне конфигурације са 6 степени слободe, који у конфигурацијама од к3.1 до к3.5, слике 3.3 до 3.7, емулира различите машине алатке при чему је:

- за конфигурацију к3.1, слика 3.3, у којој се стандардизованим роботом вертикалне зглобне конфигурације са 6 степени слободe емулира специјализовани робот за обраду вертикалне зглобне конфигурације са 5 степени слободe, угао унутрашње координате $\theta_6 = 0^\circ$ (блокирана оса),
- за конфигурацију к3.2, слика 3.4, у којој се стандардизованим роботом вертикалне зглобне конфигурације са 6 степени слободe емулира стандардизовани робот за обраду вертикалне зглобне конфигурације са 5 степени слободe, угао унутрашње координате $\theta_4 = 0^\circ$ (блокирана оса),
- за конфигурацију к3.5, слика 3.7, у којој се стандардизованим роботом вертикалне зглобне конфигурације са 6 степени слободe емулира специфична конфигурација робота са 5 степени слободe, која омогућава веће распоне углова оријентације алата A и C , угао унутрашње координате θ_4 блокиран у положају $\theta_4 = +90^\circ$ или $\theta_4 = -90^\circ$.

3.2.2 Вектори спољашњих координата

Као што је познато, *CAD/CAM* систем генерише путању алата (CL) као низ сукцесивних позиција и оријентација алата у координатном систему радног предмета $\{W\}$. Тако израчуната путања је независна од конфигурације машине. Као што је показано на сликама 3.1 до 3.7, позиција алата је описана вектором положаја врха алата T у координатном систему радног предмета $\{W\}$ као

$${}^W \mathbf{p}_T = [x_{TW} \ y_{TW} \ z_{TW}]^T \quad (3.3)$$

док је оријентација алата описана јединичним вектором осе алата, као

$${}^W \mathbf{k}_T = [k_{TWx} \ k_{TWy} \ k_{TWz}]^T \quad (3.4)$$

Како осе координатних система $\{M\}$ и $\{W\}$ не морају бити паралелне, вектор положаја врха алата и јединични вектор осе алата могу бити изражени у координатном систему $\{M\}$ као

$${}^M \mathbf{p}_T = [x_M \ y_M \ z_M]^T = {}^M \mathbf{p}_{OW} + {}^M \mathbf{R} \cdot {}^W \mathbf{p}_T \quad (3.5)$$

$${}^M \mathbf{k}_T = [k_{Tx} \ k_{Ty} \ k_{Tz}]^T = {}^M \mathbf{R} \cdot {}^W \mathbf{k}_T \quad (3.6)$$

при чему ${}^M \mathbf{p}_{OW} = [x_{OW} \ y_{OW} \ z_{OW}]^T$ представља вектор положаја координатног почетка координатног система радног предмета $\{W\}$. Одређивање вектора положаја ${}^M \mathbf{p}_{OW}$ и матрице оријентације ${}^M \mathbf{R}$ координатног система радног предмета $\{W\}$, у једначинама (3.5) и (3.6), врши се према стандардној процедури за 5-осне *CNC* машине алатке.

У овој фази развоја система је предвиђено да координатни системи $\{M\}$ и $\{W\}$ буду паралелни, тако да једначине (3.5) и (3.6) постају

$${}^M \mathbf{p}_T = [x_M \ y_M \ z_M]^T = {}^M \mathbf{p}_{OW} + {}^W \mathbf{p}_T \quad (3.7)$$

и

$${}^M \mathbf{k}_T = [k_{Tx} \ k_{Ty} \ k_{Tz}]^T = {}^W \mathbf{k}_T \quad (3.8)$$

Да би вектор спољашњих координата био комплетан, неопходно је одредити и углове оријентације (A, B) и (A, C) , који дефинишу оријентацију осе алата z_T , за све разматране конфигурације са слика 3.1 до 3.7.

Позиција и оријентација једног координатног система релативно у односу на други, у овом случају координатног система алата $\{T\}$ у односу на координатни систем основе робота, односно емулиране машине $\{M\}$, као што је познато, могу бити изражене матрицама хомогених трансформација 4 x 4 [71-73] као

$${}^M \mathbf{T} = \begin{bmatrix} {}^M \mathbf{R} & {}^M \mathbf{p}_T \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.9)$$

при чему матрица ротације ${}^M \mathbf{R}$ одређује оријентацију, а вектор ${}^M \mathbf{p}_T$ положај

координатног почетка координатног система $\{T\}$ у односу на координатни систем $\{M\}$. За постизање жељене оријентације осе алата z_T у односу на координатни систем $\{M\}$, координатни систем $\{T\}$ се, за различите типове емулираних машина, мора на одговарајући начин ротирати око оса координатног система $\{M\}$, као што је даље приказано.

Углови оријентације алата емулиране вертикалне глодалице типа (X, Y, Z, A, B)

Да би се оса алата z_T координатног система $\{T\}$ довела у жељену угаону позицију у односу на референтни координатни систем $\{M\}$, координатни систем алата $\{T\}$ се мора прво ротирати око осе X_M за угао A , а затим око осе Y_M за угао B , као што је прописано конвенцијом за вертикалне 5-осне машине алатке типа (X, Y, Z, A, B) , код којих оријентацију алата остварују осе ротација главног вретена [24, 71, 74]. Матрица ротације ${}^M_T\mathbf{R}$ се изводи као

$${}^M_T\mathbf{R} = \mathbf{R}_{Y_{M,B}} \cdot \mathbf{R}_{X_{M,A}} = \begin{bmatrix} cB & sAsB & cAsB \\ 0 & cA & -sA \\ -sB & sAcB & cAcB \end{bmatrix} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} \\ i_{Ty} & j_{Ty} & k_{Ty} \\ i_{Tz} & j_{Tz} & k_{Tz} \end{bmatrix} \quad (3.10)$$

где су

$$\mathbf{R}_{Y_{M,B}} = \begin{bmatrix} cB & 0 & sB \\ 0 & 1 & 0 \\ -sB & 0 & cB \end{bmatrix} \quad \text{и} \quad \mathbf{R}_{X_{M,A}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cA & -sA \\ 0 & sA & cA \end{bmatrix} \quad (3.11)$$

основне матрице ротације [73], а „ c “ и „ s “ представљају косинусне и синусне функције углова оријентације алата. С обзиром на то да је од интереса само оријентација осе алата z_T , која је у CL датотеци дефинисана јединичним вектором ${}^M\mathbf{k}_T = [k_{Tx} \ k_{Ty} \ k_{Tz}]^T$, изједначавањем одговарајућих чланова матрица ${}^M_T\mathbf{R}$ из једначине (3.10) се према [71, 74] углови A и B се могу одредити као

$$A = \text{Atan2} \left(-k_{Ty}, \sqrt{1 - k_{Ty}^2} \right) \quad (3.12)$$

$$B = \text{Atan2} \left(\frac{k_{Tx}}{cA}, \frac{k_{Tz}}{cA} \right) \quad (3.13)$$

Иако у једначини (3.12) постоји и друго решење, коришћењем позитивне поткорене вредности, увек се рачуна решење за $-90^\circ \leq A \leq 90^\circ$ [71, 74]. Сада

се вектор спољашњих координата, за конфигурације к1.1, слика 3.1, к3.1, слика 3.3 и к3.3, слика 3.5, изражава као

$$\mathbf{x} = [x_M \ y_M \ z_M \ A \ B]^T \quad (3.14)$$

При овоме се напомиње да у конфигурацији к3.3, слика 3.5, када робот са 6 степени слободе изводи 5-осну обраду, угао оријентације алата C , који није садржан у вектору спољашњих координата, једначина (3.14), се одржава константним, тј. $C = 0^\circ$.

Углови оријентације алата емулиране хоризонталне глодалице типа (X, Y, Z, A, B)

Да би се оса алата z_T координатног система $\{T\}$ довела у жељену угаону позицију у односу на референтни координатни систем $\{M\}$, координатни систем алата $\{T\}$ се мора прво ротирати око осе Y_M за угао B , а затим око осе X_M за угао A , као што је прописано конвенцијом за хоризонталне 5-осне машине алатке типа (X, Y, Z, A, B) , код којих оријентацију алата остварују осе ротација главног вретена. Матрица ротације ${}^M_T\mathbf{R}$ се изводи као

$${}^M_T\mathbf{R} = \mathbf{R}_{X_{M,A}} \cdot \mathbf{R}_{Y_{M,B}} = \begin{bmatrix} cB & 0 & sB \\ sAsB & cA & -sAcB \\ -cAsB & sA & cAcB \end{bmatrix} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} \\ i_{Ty} & j_{Ty} & k_{Ty} \\ i_{Tz} & j_{Tz} & k_{Tz} \end{bmatrix} \quad (3.15)$$

где су основне матрице ротације дате у једначини (3.11). С обзиром на то да је од интереса само оријентација осе алата z_T , која је у CL датотеци дефинисана јединичним вектором ${}^M\mathbf{k}_T = [k_{Tx} \ k_{Ty} \ k_{Tz}]^T$, изједначавањем одговарајућих чланова матрица ${}^M_T\mathbf{R}$ из једначине (3.15) се према [71, 74] углови A и B се могу одредити као

$$B = \text{Atan2}\left(k_{Tx}, \sqrt{1 - k_{Tx}^2}\right) \quad (3.16)$$

$$A = \text{Atan2}\left(\frac{-k_{Ty}}{cB}, \frac{k_{Tz}}{cB}\right) \quad (3.17)$$

Иако у једначини (3.16) постоји и друго решење, коришћењем позитивне поткорене вредности, увек се рачуна решење за $-90^\circ \leq B \leq 90^\circ$ [71, 74]. Сада се вектор спољашњих координата, за конфигурације к2.1, слика 3.2, к3.2, слика 3.4, и к3.4, слика 3.6, изражава као

$$\mathbf{x} = [x_M \ y_M \ z_M \ A \ B]^T \quad (3.18)$$

При овоме се напомиње да у конфигурацији к3.4, слика 3.6, када робот са 6 степени слободе изводи 5-осну обраду, угао оријентације алата C , који није садржан у вектору спољашњих координата, једначина (3.18), се одржава константним, тј. $C = 0^\circ$.

Углови оријентације алата емулиране вертикалне глодалице типа (X, Y, Z, A, C)

Да би се оса алата z_T координатног система $\{T\}$ довела у жељену угаону позицију у односу на референтни координатни систем $\{M\}$, координатни систем алата $\{T\}$ се мора прво ротирати око осе X_M за угао A , а затим око осе Z_M за угао C , као што је прописано конвенцијом за вертикалне 5-осне машине алатке типа (X, Y, Z, A, C) , код којих оријентацију алата остварују осе ротација главног вретена. Матрица ротације ${}^M_T\mathbf{R}$ се изводи као

$${}^M_T\mathbf{R} = \mathbf{R}_{Z_{M,C}} \cdot \mathbf{R}_{X_{M,A}} = \begin{bmatrix} cC & -cAsC & sAsC \\ sC & cAcC & -sAcC \\ 0 & sA & cA \end{bmatrix} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} \\ i_{Ty} & j_{Ty} & k_{Ty} \\ i_{Tz} & j_{Tz} & k_{Tz} \end{bmatrix} \quad (3.19)$$

где су основне матрице ротације

$$\mathbf{R}_{Z_{M,C}} = \begin{bmatrix} cC & -sC & 0 \\ sC & cC & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{и} \quad \mathbf{R}_{X_{M,A}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cA & -sA \\ 0 & sA & cA \end{bmatrix} \quad (3.20)$$

С обзиром на то да је од интереса само оријентација осе алата z_T , која је дефинисана у CL датотеци јединичним вектором ${}^M\mathbf{k}_T = [k_{Tx} \ k_{Ty} \ k_{Tz}]^T$, изједначавањем одговарајућих чланова матрица ${}^M_T\mathbf{R}$ из једначине (3.15) се према [71, 74] углови A и C се могу одредити као

$$A = \text{Atan2} \left(\sqrt{1 - k_{Tz}^2}, k_{Tz} \right) \quad (3.21)$$

$$C = \text{Atan2} \left(\frac{k_{Tx}}{sA}, \frac{-k_{Ty}}{sA} \right) \quad (3.22)$$

Иако у једначини (3.21) постоји и друго решење, коришћењем позитивне поткорене вредности, увек се рачуна решење за $-90^\circ \leq A \leq 90^\circ$ [71, 74]. Сада се вектор спољашњих координата, за конфигурацију к3.5, слика 3.7, изражава као

$$\mathbf{x} = [x_M \ y_M \ z_M \ A \ C]^T \quad (3.23)$$

3.3 Директни кинематички проблем елементарних ћелија на бази робота као емулираних 5-осних машина алатки

Елементарна обрадна ћелија, која укључује робота вертикалне зглобне конфигурације са 5 или 6 степени слободе и непокретан радни сто, може емулирати 5-осне хоризонталне и вертикалне глодалице, код којих оријентацију алата остварују осе ротација главног вретена. Да би се за програмирање овако емулираних 5-осних машина алатки могли користити постојећи *CAD/CAM* системи, одговарајући координатни системи су усвојени према стандардима за машине алатке, као што је показано на сликама 3.1 до 3.7. Основа за генерализовано решење директног кинематичког проблема (*DKP*), а касније и инверзног кинематичког проблема елементарних ћелија на бази робота вертикалне зглобне конфигурације, односно емулираних машина алатки, чине кинематички модели:

- Стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе;
- Специјализованог робота вертикалне зглобне конфигурације са 5 степени слободе;
- Стандардизованог робота вертикалне зглобне конфигурације са 5 степени слободе;
- Стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе и блокираном осом θ_4 у позицији $+90^\circ$ или -90° .

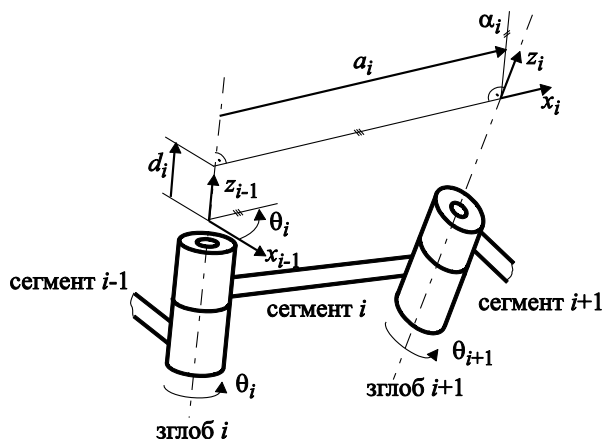
3.3.1 Директни кинематички проблем робота вертикалне зглобне конфигурације са 5 и 6 степени слободе

Приликом кинематичког моделирања робота, примењена је *Denavit-Hartenberg (D-H)* конвенција [71, 72]. Као што се види са слика 3.1 до 3.7, работи се састоје од пет или шест покретних сегмената међусобно повезаних са пет, односно шест ротационих зглобова. Први покретни сегмент је повезан за непокретну основу, док је за последњи сегмент везана обрадна јединица (моторно вретено) са алатом. Да би се извршила кинематичка анализа, најпре се сваком сегменту робота додељује координатни систем. Релативан положај између ових координатних система се може описати помоћу

хомогених трансформација. Хомогена трансформација која описује положај једног сегмента у односу на други се традиционално обележава као A матрица. Матрица ${}^{i-1}_iA$ означава $D-H$ трансформациону матрицу која описује положај координатног система (i) у односу на координатни систем ($i - 1$) и представљена је на следећи начин

$${}^{i-1}_iA = \begin{bmatrix} {}^{i-1}n_i & {}^{i-1}o_i & {}^{i-1}a_i & {}^{i-1}p_{oi} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.24)$$

На слици 3.8 је приказан релативан положај између два суседна сегмента са придруженим координатним системима. $D-H$ модел користи четири параметара ($a_i, \alpha_i, d_i, \theta_i$) за опис релативних трансформација између сегмената ($i - 1$) и (i). Први параметар a_i представља растојање дуж заједничке нормале између оса зглобова. Други параметар α_i представља угао између оса зглобова. Преостала два параметра, који одређују релативан положај између два суседна сегмента, представљају дужину d_i и угао θ_i .



Слика 3.8 Придруживање координатних система сегментима манипулатора и њихови кинематички параметри

Након придруживања координатних система сегментима, хомогена матрица трансформације ${}^{i-1}_iA$, која описује положај i -тог сегмента у односу на претходни сегмент, се изводи релативним трансформацијама као

$${}^{i-1}_iA = Rot(z_{i-1}, \theta_i) \cdot Trans(0,0, d_i) \cdot Trans(a_i, 0,0) \cdot Rot(x_i, \alpha_i) =$$

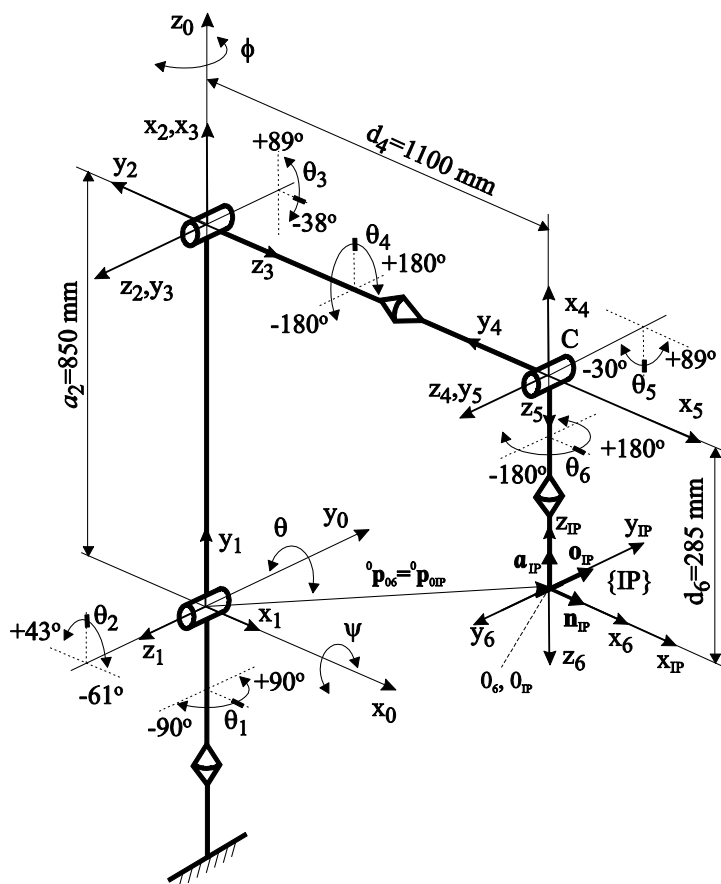
$$\begin{bmatrix} c\theta_i & -c\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\alpha_i c\theta_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.25)$$

Даље је приказан генерализован приступ кинематичког моделирања у коме се полази од добро познатог кинематичког модела стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе, али тако да обухвати и кинематичке моделе два основна типа робота вертикалне зглобне конфигурације са 5 степени слободе приказане на сликама 3.1 и 3.2.

Директни кинематички проблем стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе

Кинематички модел експерименталног стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе, са слике 2.11, са $D-H$ координатним системима $O_i x_i y_i z_i$, који су придружени сегментима робота $i = 1, 2, \dots, 6$, приказан је на слици 3.9.

За овако придружене координатне системе сегментима робота и усвојен референтни положај, слика 3.9, $D-H$ кинематички параметри су дати у табели 3.1.



Слика 3.9 $D-H$ координатни системи робота вертикалне зглобне конфигурације са 6 степени слободе

Ограничења кретања зглобова робота су специфицирана у односу на усвојени референтни положај робота и представљају мање опсеге кретања него што су стварни опсези кретања робота. Усвојени опсези кретања зглобова робота су показани на слици 3.9 и омогућавају само једно решење инверзног кинематичког проблема које је логично при обради овим роботом, а такође омогућавају и избегавање сингуларних конфигурација робота.

Табела 3.1. D-H кинематички параметри

$link\ i$	$\alpha_i [^\circ]$	$a_i [mm]$	$d_i [mm]$	$\theta_i [^\circ]$
1	90	0	0	θ_1
2	0	a_2	0	$\theta_2 + 90$
3	90	0	0	θ_3
4	-90	0	d_4	$-\theta_4$
5	90	0	0	$\theta_5 - 90$
6	0	0	d_6	$-\theta_6$

Заменом D-H параметара из табеле 3.1 у једначину (3.25) добијају се матрице

${}^{i-1}_i\mathbf{A}$, $i = 1, 2, \dots, 6$ као

$$\begin{aligned}
 {}^0_1\mathbf{A} &= \begin{bmatrix} c\theta_1 & 0 & s\theta_1 & 0 \\ s\theta_1 & 0 & -c\theta_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^1_2\mathbf{A} &= \begin{bmatrix} -s\theta_2 & -c\theta_2 & 0 & -a_2s\theta_2 \\ c\theta_2 & -s\theta_2 & 0 & a_2c\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^2_3\mathbf{A} &= \begin{bmatrix} c\theta_3 & 0 & s\theta_3 & 0 \\ s\theta_3 & 0 & -c\theta_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^3_4\mathbf{A} &= \begin{bmatrix} c\theta_4 & 0 & s\theta_4 & 0 \\ -s\theta_4 & 0 & c\theta_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^4_5\mathbf{A} &= \begin{bmatrix} s\theta_5 & -c\theta_5 & 0 & 0 \\ -c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^5_6\mathbf{A} &= \begin{bmatrix} c\theta_6 & s\theta_6 & 0 & 0 \\ -s\theta_6 & c\theta_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{3.26}$$

Даље се, као што је добро познато [71, 72], позиција и оријентација координатног система додатног интерфејса $\{IP\}$, ${}^0\mathbf{p}_{0IP}$ и ${}^0_{IP}\mathbf{R}$, у односу на координатни систем основе робота (x_0, y_0, z_0) , за дати вектор унутрашњих координата $\boldsymbol{\theta} = [\theta_i]^T$, $i = 1, 2, \dots, 6$ и специфициране кинематичке параметре, може се изразити на следећи начин

$$\begin{aligned}
 {}^0_I P \mathbf{T} &= {}^0_1 \mathbf{A}_2 {}^1_2 \mathbf{A}_3 {}^2_3 \mathbf{A}_4 {}^3_4 \mathbf{A}_5 {}^4_5 \mathbf{A}_6 {}^5_6 \mathbf{A}_{IP} \mathbf{T} = {}^0_6 \mathbf{T} {}^6_{IP} \mathbf{T} = {}^0_{IP} \mathbf{T}(\theta_1, \dots, \theta_6) = \\
 \begin{bmatrix} & {}^0_{IP} \mathbf{R} & & {}^0 \mathbf{p}_{0IP} \\ 0 & 0 & 0 & 1 \end{bmatrix} &= \begin{bmatrix} n_{IPx} & o_{IPx} & a_{IPx} & p_{IPx} \\ n_{IPy} & o_{IPy} & a_{IPy} & p_{IPy} \\ n_{IPz} & o_{IPz} & a_{IPz} & p_{IPz} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.27)
 \end{aligned}$$

где је

$${}^0_6 \mathbf{T} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.28)$$

док је

$${}^6_{IP} \mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.29)$$

хомогена матрична трансформација која описује положај и оријентацију уведеног координатног система додатног интерфејса $\{IP\}$ у односу на координатни систем (x_6, y_6, z_6) , слика 3.9. Заменом матрица из једначина 3.26 и 3.29 у једначину 3.27 се добијају чланови матрице ${}^0_{IP} \mathbf{T}(\theta_1, \dots, \theta_6)$ као

$$\begin{aligned}
 n_{IPx} &= -s\theta_1(s\theta_4 \cdot s\theta_5 \cdot c\theta_6 + c\theta_4 \cdot s\theta_6) + \\
 &\quad c\theta_1(c\theta_{23} \cdot c\theta_5 \cdot c\theta_6 + s\theta_{23}(s\theta_4 \cdot s\theta_6 - c\theta_4 \cdot s\theta_5 \cdot c\theta_6)) \\
 n_{IPy} &= s\theta_6(s\theta_1 \cdot s\theta_{23} \cdot s\theta_4 + c\theta_1 \cdot c\theta_4) + \quad (3.30) \\
 &\quad c\theta_6(s\theta_1 \cdot c\theta_{23} \cdot c\theta_5 + s\theta_5(c\theta_1 \cdot s\theta_4 - s\theta_1 \cdot s\theta_{23} \cdot c\theta_4))
 \end{aligned}$$

$$\begin{aligned}
 n_{IPz} &= s\theta_{23} \cdot c\theta_5 \cdot c\theta_6 + c\theta_{23}(c\theta_4 \cdot s\theta_5 \cdot c\theta_6 - s\theta_4 \cdot s\theta_6) \\
 o_{IPx} &= c\theta_6(c\theta_1 \cdot s\theta_{23} \cdot s\theta_4 - s\theta_1 \cdot c\theta_4) - \\
 &\quad s\theta_6(c\theta_1 \cdot c\theta_{23} \cdot c\theta_5 - s\theta_5(s\theta_1 \cdot s\theta_4 + c\theta_1 \cdot s\theta_{23} \cdot c\theta_4)) \\
 o_{IPy} &= c\theta_6(s\theta_1 \cdot s\theta_{23} \cdot s\theta_4 + c\theta_1 \cdot c\theta_4) - \quad (3.31) \\
 &\quad s\theta_6(s\theta_1 \cdot c\theta_{23} \cdot c\theta_5 + s\theta_5(c\theta_1 \cdot s\theta_4 - s\theta_1 \cdot s\theta_{23} \cdot c\theta_4)) \\
 o_{IPz} &= -s\theta_{23} \cdot c\theta_5 \cdot s\theta_6 - c\theta_{23}(c\theta_4 \cdot s\theta_5 \cdot s\theta_6 + s\theta_4 \cdot c\theta_6)
 \end{aligned}$$

$$\begin{aligned}
 a_{IPx} &= -s\theta_1 \cdot s\theta_4 \cdot c\theta_5 - c\theta_1 c\theta_3 (c\theta_2 \cdot s\theta_5 + s\theta_2 \cdot c\theta_4 \cdot c\theta_5) + \\
 &\quad c\theta_1 s\theta_3 (s\theta_2 \cdot s\theta_5 - c\theta_2 \cdot c\theta_4 \cdot c\theta_5) \\
 a_{IPy} &= -s\theta_1 \cdot c\theta_{23} \cdot s\theta_5 - c\theta_5 \cdot s\theta_1 \cdot s\theta_2 \cdot c\theta_3 \cdot c\theta_4 - \\
 &\quad c\theta_5 \cdot s\theta_1 \cdot c\theta_2 \cdot s\theta_3 \cdot c\theta_4 + c\theta_5 \cdot c\theta_1 \cdot s\theta_4
 \end{aligned} \tag{3.32}$$

$$a_{IPz} = -s\theta_2 (c\theta_3 \cdot s\theta_5 + s\theta_3 \cdot c\theta_4 \cdot c\theta_5) - c\theta_2 (s\theta_3 \cdot s\theta_5 - c\theta_3 \cdot c\theta_4 \cdot c\theta_5)$$

$$\begin{aligned}
 p_{IPx} &= -c\theta_1 \cdot s\theta_2 \cdot a_2 + c\theta_1 \cdot c\theta_{23} \cdot d_4 - a_{IPx} \cdot a_5 \\
 p_{IPy} &= -s\theta_1 \cdot s\theta_2 \cdot a_2 + s\theta_1 \cdot c\theta_{23} \cdot d_4 - a_{IPy} \cdot a_5 \\
 p_{IPz} &= c\theta_2 \cdot a_2 + s\theta_{23} \cdot d_4 - a_{IPz} \cdot a_5
 \end{aligned} \tag{3.33}$$

где је $\theta_{ij} = \theta_i + \theta_j$.

Овако израчунат вектор положаја ${}^0\mathbf{p}_{0IP}$, једначина 3.33, и матрица оријентације ${}^0\mathbf{R}_{IP}$, једначине 3.30 до 3.32, представљају решење директног кинематичког проблема стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе.

Директни кинематички проблем специјализованог робота вертикалне зглобне конфигурације са 5 степени слободе

Кинематички модел специјализованог робота вертикалне зглобне конфигурације са 5 степени слободе са слике 2.13, са $D-H$ координатним системима $O_i x_i y_i z_i$, који су придружени сегментима робота $i = 1, 2, \dots, 5$, приказан је на слици 3.10.

Позиција и оријентација координатног система додатног интерфејса $\{IP_v\}$, ${}^0\mathbf{p}_{0IP_v}$ и ${}^0\mathbf{R}_{IP_v}$, у односу на координатни систем основе робота (x_0, y_0, z_0) , за дати вектор унутрашњих координата $\boldsymbol{\theta} = [\theta_i]^T$, $i = 1, 2, \dots, 5$ и специфициране кинематичке параметре, може се изразити на следећи начин

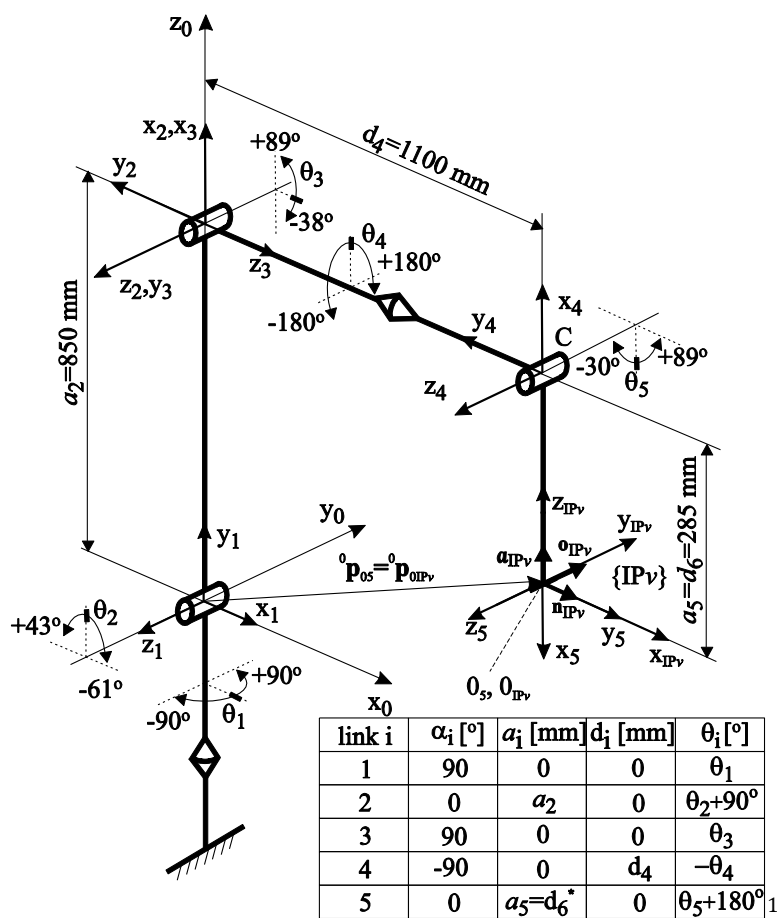
$$\begin{aligned}
 {}^{0}_{IP_v}\mathbf{T} &= {}^0\mathbf{A}_1^1 \mathbf{A}_2^2 \mathbf{A}_3^3 \mathbf{A}_4^4 \mathbf{A}_5^5 \mathbf{A}_{IP_v}^5 \mathbf{T} = {}^5_5\mathbf{T} {}^{5}_{IP_v}\mathbf{T} = {}^{0}_{IP_v}\mathbf{T}(\theta_1, \dots, \theta_5) = \\
 \begin{bmatrix} & & & & \\ & {}^0\mathbf{R}_{IP_v} & & & \\ 0 & 0 & 0 & 1 & \end{bmatrix} &= \begin{bmatrix} n_{IP_v x} & o_{IP_v x} & a_{IP_v x} & p_{IP_v x} \\ n_{IP_v y} & o_{IP_v y} & a_{IP_v y} & p_{IP_v y} \\ n_{IP_v z} & o_{IP_v z} & a_{IP_v z} & p_{IP_v z} \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{3.34}$$

где је

$${}^0_5T = \begin{bmatrix} n_{x_v} & o_{x_v} & a_{x_v} & p_{x_v} \\ n_{y_v} & o_{y_v} & a_{y_v} & p_{y_v} \\ n_{z_v} & o_{z_v} & a_{z_v} & p_{z_v} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.35)$$

док је

$${}^{IP_v}_5T = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.36)$$



Слика 3.10 D-H координатни системи и кинематички параметри специјализованог робота за обраду са 5 степени

* $a_5 = d_6$ је усвојено да би се омогућила директно поређење изведених једначина DKP специјализованог робота за обраду са 5 степени слободе и интегрисаним моторним вretenом са једначинама DKP стандардизованог робота са 6 степени слободе

хомогена матрична трансформација која описује положај и оријентацију додатног интерфејса $\{IP_v\}$ у односу на координатни систем (x_5, y_5, z_5) , слика 3.1. Заменом A матрица, које се добијају из $D-H$ табеле на слици 3.10 и једначине 3.36 у једначину 3.34 се добијају чланови матрице ${}_{IP_v}^0\mathbf{T}(\theta_1, \dots, \theta_5)$ као

$$\begin{aligned} n_{IP_vx} &= c\theta_1 \cdot c\theta_{23} \cdot c\theta_5 - s\theta_5(c\theta_1 \cdot s\theta_{23} \cdot c\theta_4 + s\theta_1 \cdot s\theta_4) \\ n_{IP_vy} &= s\theta_1 \cdot c\theta_{23} \cdot c\theta_5 - s\theta_5(s\theta_1 \cdot s\theta_{23} \cdot c\theta_4 - c\theta_1 \cdot s\theta_4) \\ n_{IP_vz} &= c\theta_5 \cdot s\theta_{23} + c\theta_{23} \cdot c\theta_4 \cdot s\theta_5 \end{aligned} \quad (3.37)$$

$$\begin{aligned} o_{IP_vx} &= c\theta_1 \cdot s\theta_{23} \cdot s\theta_4 - s\theta_1 \cdot c\theta_4 \\ o_{IP_vy} &= s\theta_1 \cdot s\theta_{23} \cdot s\theta_4 + c\theta_1 \cdot c\theta_4 \\ o_{IP_vz} &= -c\theta_{23} \cdot s\theta_4 \end{aligned} \quad (3.38)$$

$$\begin{aligned} a_{IP_vx} &= -s\theta_1 \cdot s\theta_4 \cdot c\theta_5 - c\theta_1(s\theta_{23} \cdot c\theta_4 \cdot c\theta_5 + c\theta_{23} \cdot s\theta_5) \\ a_{IP_vy} &= -s\theta_1 \cdot c\theta_{23} \cdot s\theta_5 - c\theta_5(s\theta_1 \cdot c\theta_4 \cdot s\theta_{23} - c\theta_1 \cdot s\theta_4) \\ a_{IP_vz} &= c\theta_{23} \cdot c\theta_4 \cdot c\theta_5 - s\theta_{23} \cdot s\theta_5 \end{aligned} \quad (3.39)$$

$$\begin{aligned} p_{IP_vx} &= -c\theta_1 \cdot s\theta_2 \cdot a_2 + c\theta_1 \cdot c\theta_{23} \cdot d_4 - a_{IP_vx} \cdot a_5 \\ p_{IP_vy} &= -s\theta_1 \cdot s\theta_2 \cdot a_2 + s\theta_1 \cdot c\theta_{23} \cdot d_4 - a_{IP_vy} \cdot a_5 \\ p_{IP_vz} &= c\theta_2 \cdot a_2 + s\theta_{23} \cdot d_4 - a_{IP_vz} \cdot a_5 \end{aligned} \quad (3.40)$$

Овако израчунат вектор положаја ${}^0\mathbf{p}_{0IP_v}$, једначина 3.40, и матрица оријентације ${}_{IP_v}^0\mathbf{R}$, једначине 3.37 до 3.39, представљају решење директног кинематичког проблема специјализованог робота вертикалне зглобне конфигурације са 5 степени слободе.

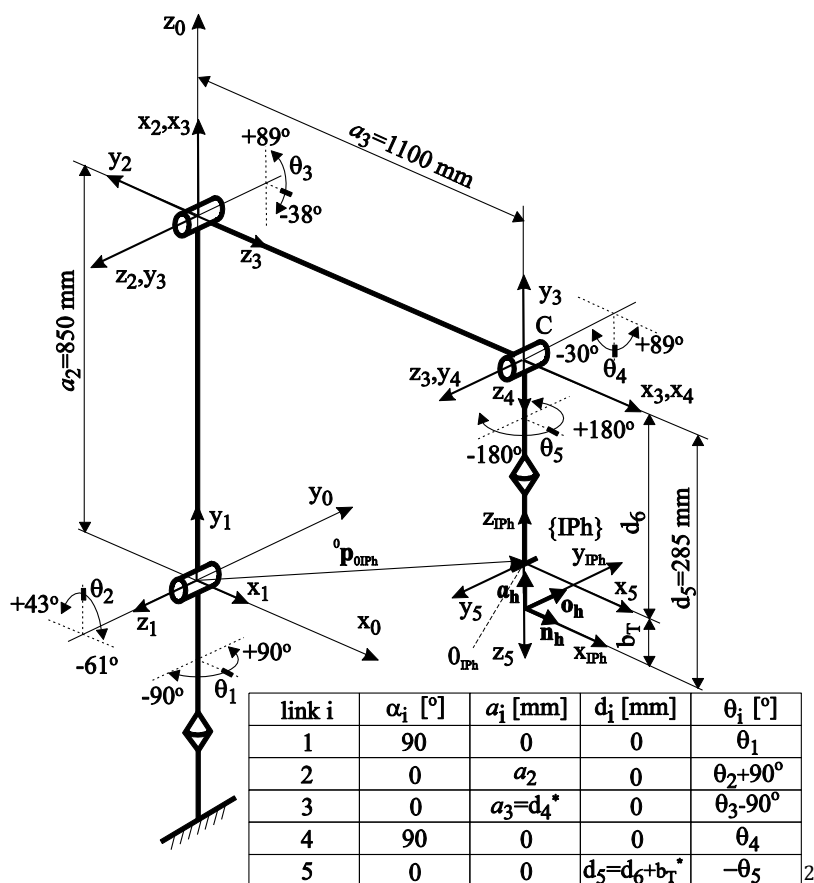
Као што је наглашено, оваквим приступом кинематичког моделирања се показује да је решење директног кинематичког проблема специјализованог робота са 5 степени слободе са слике 3.10, једначине 3.37 до 3.40, идентично решењу директног кинематичког проблема стандардизованог робота са 6 степени слободе, када се у једначине 3.30 до 3.33 уврсти да је $\theta_6 = 0^\circ$, односно да је

$${}_{IP}^0\mathbf{T}(\theta_1, \dots, \theta_5, \theta_6 = 0^\circ(\text{блокирана})) = {}_{IP_v}^0\mathbf{T}(\theta_1, \dots, \theta_5) \quad (3.41)$$

Директни кинематички проблем стандардизованог робота вертикалне зглобне конфигурације са 5 степени слободе

Кинематички модел стандардизованог робота вертикалне зглобне конфигурације са 5 степени слободе са слике 2.14, са *D-H* координатним системима $O_i x_i y_i z_i$, који су придружени сегментима робота $i = 1, 2, \dots, 5$, приказан је на слици 3.11.

Позиција и оријентација координатног система додатног интерфејса $\{IP_h\}$, ${}^0 p_{0IP_h}$ и ${}_{IP_h}^0 R$, у односу на координатни систем основе робота (x_0, y_0, z_0) , за дати вектор унутрашњих координата $\theta = [\theta_i]^T$, $i = 1, 2, \dots, 5$ и специфициране кинематичке параметре, може се изразити на следећи начин



Слика 3.11 *D-H* координатни системи и кинематички параметри стандардизованог робота за обраду са 5 степени слободе

* $a_3 = d_4$ и $d_5 = d_6 + b_T$ је усвојено да би се омогућила директно поређење изведених једначина *DKP* стандардизованог робота са 5 степени слободе са једначинама *DKP* стандардизованог робота са 6 степени слободе

$$\begin{aligned}
 {}_{IP_h}^0\mathbf{T} &= {}_1^0\mathbf{A}_2^1\mathbf{A}_3^2\mathbf{A}_4^3\mathbf{A}_5^4\mathbf{A}_{IP_h}^5\mathbf{T} = {}_5^0\mathbf{T} \quad {}_{IP_h}^5\mathbf{T} = {}_{IP_h}^0\mathbf{T}(\theta_1, \dots, \theta_5) = \\
 \begin{bmatrix} & & & \\ & {}_{IP_h}^0\mathbf{R} & & \\ 0 & 0 & 0 & 1 \end{bmatrix} &= \begin{bmatrix} n_{IP_h x} & o_{IP_h x} & a_{IP_h x} & p_{IP_h x} \\ n_{IP_h y} & o_{IP_h y} & a_{IP_h y} & p_{IP_h y} \\ n_{IP_h z} & o_{IP_h z} & a_{IP_h z} & p_{IP_h z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.42)
 \end{aligned}$$

где је

$${}_5^0\mathbf{T} = \begin{bmatrix} n_{x_h} & o_{x_h} & a_{x_h} & p_{x_h} \\ n_{y_h} & o_{y_h} & a_{y_h} & p_{y_h} \\ n_{z_h} & o_{z_h} & a_{z_h} & p_{z_h} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.43)$$

док је

$${}_{IP_h}^5\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.44)$$

хомогена матрична трансформација која описује положај и оријентацију уведеног координатног система додатног интерфејса $\{IP_h\}$ у односу на координатни систем (x_5, y_5, z_5) , слика 3.2. Заменом \mathbf{A} матрица, које се добијају из $D-H$ табеле на слици 3.11 и једначине 3.44 у једначину 3.42 се добијају чланови матрице ${}_{IP_h}^0\mathbf{T}(\theta_1, \dots, \theta_5)$ као

$$\begin{aligned}
 n_{IP_h x} &= c\theta_5(c\theta_1 \cdot c\theta_{23} \cdot c\theta_4 - c\theta_1 \cdot s\theta_{23} \cdot s\theta_4) - s\theta_1 \cdot s\theta_5 \\
 n_{IP_h y} &= c\theta_5(s\theta_1 \cdot c\theta_{23} \cdot c\theta_4 - s\theta_1 \cdot s\theta_{23} \cdot s\theta_4) + c\theta_1 \cdot s\theta_5 \\
 n_{IP_h z} &= c\theta_5(s\theta_{23} \cdot c\theta_4 + c\theta_{23} \cdot s\theta_4)
 \end{aligned} \quad (3.45)$$

$$\begin{aligned}
 o_{IP_h x} &= -s\theta_5(c\theta_1 \cdot c\theta_{23} \cdot c\theta_4 - c\theta_1 \cdot s\theta_{23} \cdot s\theta_4) - s\theta_1 \cdot c\theta_5 \\
 o_{IP_h y} &= -s\theta_5(s\theta_1 \cdot c\theta_{23} \cdot c\theta_4 - s\theta_1 \cdot s\theta_{23} \cdot s\theta_4) + c\theta_1 \cdot c\theta_5 \\
 o_{IP_h z} &= -s\theta_5(s\theta_{23} \cdot c\theta_4 + c\theta_{23} \cdot s\theta_4)
 \end{aligned} \quad (3.46)$$

$$\begin{aligned}
 a_{IP_h x} &= -c\theta_1 \cdot s\theta_{23} \cdot c\theta_4 - c\theta_1 \cdot c\theta_{23} \cdot s\theta_4 \\
 a_{IP_h y} &= -s\theta_1 \cdot s\theta_{23} \cdot c\theta_4 - s\theta_1 \cdot c\theta_{23} \cdot s\theta_4 \\
 a_{IP_h z} &= c\theta_{23} \cdot c\theta_4 - s\theta_{23} \cdot s\theta_4
 \end{aligned} \quad (3.47)$$

$$\begin{aligned}
 p_{IP_h x} &= -c\theta_1 \cdot s\theta_2 \cdot a_2 + c\theta_1 \cdot c\theta_{23} \cdot a_3 - a_{IP_h x} \cdot d_5 \\
 p_{IP_h y} &= -s\theta_1 \cdot s\theta_2 \cdot a_2 + s\theta_1 \cdot c\theta_{23} \cdot a_3 - a_{IP_h y} \cdot d_5 \\
 p_{IP_h z} &= c\theta_2 \cdot a_2 + s\theta_{23} \cdot a_3 - a_{IP_h z} \cdot d_5
 \end{aligned} \tag{3.48}$$

Овако израчунат вектор положаја ${}^0\mathbf{p}_{0IP_h}$, једначина 3.48, и матрица оријентације ${}_{IP_h}^0\mathbf{R}$, једначине 3.45 до 3.47, представљају решење директног кинематичког проблема стандардизованог робота вертикалне зглобне конфигурације са 5 степени слободе.

Као што је наглашено, оваквим приступом кинематичког моделирања се показује да је решење директног кинематичког проблема стандардизованог робота са 5 степени слободе са слике 3.11, једначине 3.45 до 3.48, идентично решењу директног кинематичког проблема стандардизованог робота са 6 степени слободе, када се у једначине 3.30 до 3.33 уврсти да је $\theta_4 = 0^\circ$, односно да је

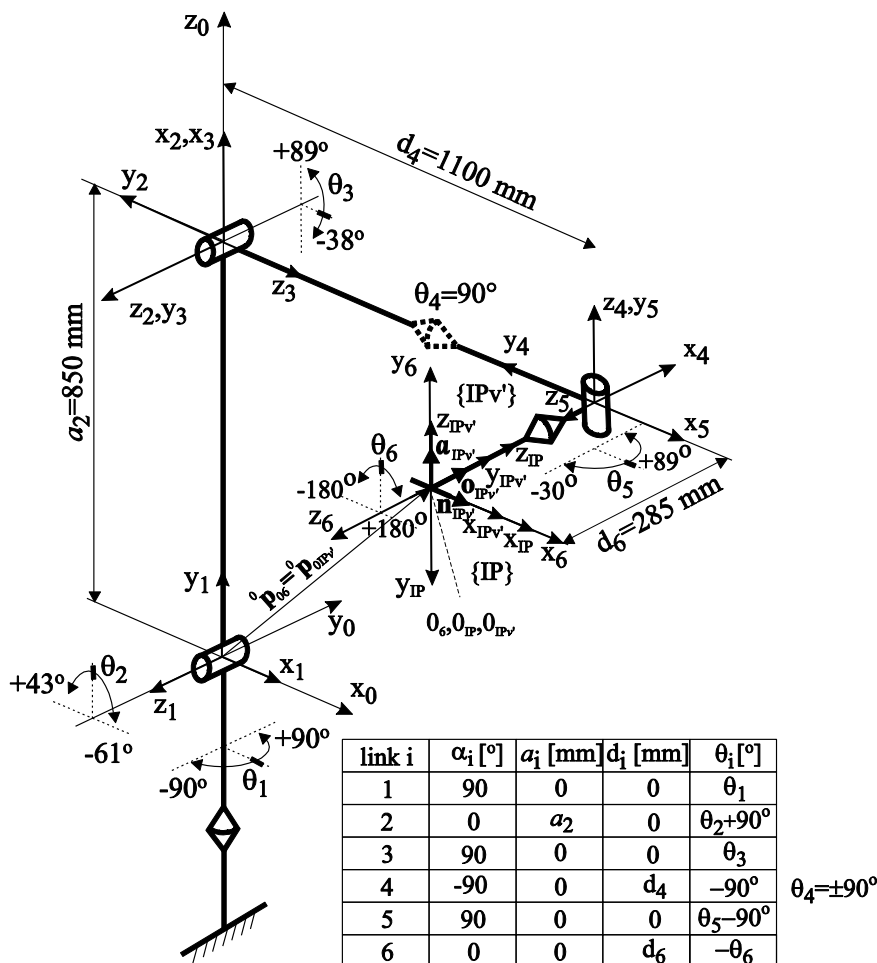
$${}_{IP}^0\mathbf{T}(\theta_1, \dots, \theta_4 = 0^\circ(\text{блокирана}), \theta_5, \theta_6,) \triangleq {}_{IP_h}^0\mathbf{T}(\theta_1, \dots, \theta_5) \tag{3.49}$$

с тим што се са слика 3.9 и 3.11 види да унутрашње координате (углови) θ_4, θ_5 стандардизованог робота са 5 степени слободе одговарају угловима θ_5, θ_6 робота са 6 степени слободе.

Директни кинематички проблем емулираног робота специфичне вертикалне зглобне конфигурације са 5 степени слободе

Специфична конфигурација стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе, у којој је оса θ_4 блокирана у позицији $+90^\circ$ или -90° , је потребна за елементарну ћелију која емулира 5-осну вертикалну глодалицу (X, Y, Z, A, C) са повећаним распонем углава оријентације алата, слика 3.7. Кинематички модел оваквог робота, са $D-H$ координатним системима $O_i x_i y_i z_i$, који су придружени сегментима робота $i = 1, 2, \dots, 6$, приказан је на слици 3.12.

Позиција и оријентација координатног система додатног интерфејса $\{IP_v'\}$, ${}^0\mathbf{p}_{0IP_v'}$ и ${}_{IP_v'}^0\mathbf{R}$, у односу на координатни систем основе робота (x_0, y_0, z_0) , за дати вектор унутрашњих координата $\boldsymbol{\theta} = [\theta_i]^T$, $i = 1, 2, \dots, 6$ и специфициране кинематичке параметре, може се изразити на следећи начин



Слика 3.12 D-H координатни системи и кинематички параметри стандардизованог робота за обраду са 6 степени слободe и блокираном осом θ_4 ($\theta_4 = \pm 90^\circ$)

$${}_{IP_v}^0 T = {}_1^0 A_2^1 A_3^2 A_4^3 A_5^4 A_6^5 A_{IP_v}^6 T = {}_6^0 T {}_{IP_v}^6 T =$$

$${}_{IP_v}^0 T(\theta_1, \dots, \theta_4 = \pm 90^\circ (\text{блокирана}), \dots, \theta_6) = \begin{bmatrix} {}_{IP_v}^0 R & {}^0 p_{0IP_v'} \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} n_{IP_v'x} & o_{IP_v'x} & a_{IP_v'x} & p_{IP_v'x} \\ n_{IP_v'y} & o_{IP_v'y} & a_{IP_v'y} & p_{IP_v'y} \\ n_{IP_v'z} & o_{IP_v'z} & a_{IP_v'z} & p_{IP_v'z} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.50)$$

где је

$${}_6^0 T = \begin{bmatrix} n_{x_v'} & o_{x_v'} & a_{x_v'} & p_{x_v'} \\ n_{y_v'} & o_{y_v'} & a_{y_v'} & p_{y_v'} \\ n_{z_v'} & o_{z_v'} & a_{z_v'} & p_{z_v'} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.51)$$

док је

$${}_{IP_v'}{}^6\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.52)$$

хомогена матрична трансформација која описује положај и оријентацију уведеног координатног система додатног интерфејса $\{IP_v'\}$ у односу на координатни систем (x_6, y_6, z_6) , слика 3.7. Заменом A матрица, које се добијају из $D-H$ табеле на слици 3.12 и једначине 3.52 у једначину 3.50 се добијају чланови матрице ${}_{IP_v'}{}^0\mathbf{T}(\theta_1, \dots, \theta_4 = \pm 90^\circ (\text{блокирана}), \dots, \theta_6)$ као

$$\begin{aligned} n_{IP_v'x} &= c\theta_1(c\theta_{23} \cdot c\theta_5 \cdot c\theta_6 + s\theta_{23} \cdot s\theta_6) - s\theta_1 \cdot s\theta_5 \cdot c\theta_6 \\ n_{IP_v'y} &= s\theta_1(c\theta_{23} \cdot c\theta_5 \cdot c\theta_6 + s\theta_{23} \cdot s\theta_6) + c\theta_1 \cdot s\theta_5 \cdot c\theta_6 \\ n_{IP_v'z} &= s\theta_{23} \cdot c\theta_5 \cdot c\theta_6 - c\theta_{23} \cdot s\theta_6 \end{aligned} \quad (3.53)$$

$$\begin{aligned} o_{IP_v'x} &= -s\theta_1 \cdot c\theta_5 - c\theta_1 \cdot c\theta_{23} \cdot s\theta_5 \\ o_{IP_v'y} &= c\theta_1 \cdot c\theta_5 - s\theta_1 \cdot c\theta_{23} \cdot s\theta_5 \\ o_{IP_v'z} &= -s\theta_{23} \cdot s\theta_5 \end{aligned} \quad (3.54)$$

$$\begin{aligned} a_{IP_v'x} &= c\theta_1(c\theta_{23} \cdot c\theta_5 \cdot s\theta_6 - s\theta_{23} \cdot c\theta_6) - s\theta_1 \cdot s\theta_5 \cdot s\theta_6 \\ a_{IP_v'y} &= s\theta_6(s\theta_1 \cdot c\theta_{23} \cdot c\theta_5 + c\theta_1 \cdot s\theta_5) - s\theta_1 \cdot s\theta_{23} \cdot c\theta_6 \\ a_{IP_v'z} &= c\theta_{23} \cdot c\theta_6 + s\theta_{23} \cdot c\theta_5 \cdot s\theta_6 \end{aligned} \quad (3.55)$$

$$\begin{aligned} p_{IP_v'x} &= -c\theta_1 \cdot s\theta_2 \cdot a_2 + c\theta_1 \cdot c\theta_{23} \cdot d_4 - o_{IP_v'x} \cdot d_6 \\ p_{IP_v'y} &= -s\theta_1 \cdot s\theta_2 \cdot a_2 + s\theta_1 \cdot c\theta_{23} \cdot d_4 - o_{IP_v'y} \cdot d_6 \\ p_{IP_v'z} &= c\theta_2 \cdot a_2 + s\theta_{23} \cdot a_3 - o_{IP_v'z} \cdot d_6 \end{aligned} \quad (3.56)$$

Овако израчунат вектор положаја ${}^0\mathbf{p}_{IP_h'}$, једначина 3.56, и матрица оријентације ${}_{IP_h'}{}^0\mathbf{R}$, једначине 3.53 до 3.55, представљају решење директног кинематичког проблема стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе, који емулира специфичну конфигурацију робота са 5 степени слободе ($\theta_4 = \pm 90^\circ$).

Као што је наглашено, оваквим приступом кинематичког моделирања се

показује да је решење директног кинематичког проблема стандардизованог робота вертикалне зглобне конфигурације са 6 степени слободе, који емулира специфичну конфигурацију робота са 5 степени слободе ($\theta_4 = \pm 90^\circ$) са слике 3.12, једначине 3.53 до 3.56, идентично решењу директног кинематичког проблема стандардизованог робота са 6 степени слободе, када се у једначине 3.30 до 3.33 уврсти да је $\theta_4 = \pm 90^\circ$. С обзиром на то да је у кинематички модел робота уведен координатни систем додатног интерфејса $\{IP_v'\}$, како би идентитет решења био испуњен неопходно је увести додатну трансформациону матрицу ${}^{IP_v'}T$ као

$${}^{IP_v'}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.57)$$

тако да је

$${}^{IP_v}{}^6T = {}^6T \cdot {}^{IP_v'}T \quad (3.58)$$

што, ако се узме у обзир једначина (3.50), даје следеће

$${}^0T(\theta_1, \dots, \theta_4 = \pm 90^\circ(\text{блокирана}), \theta_5, \theta_6,) = {}^{IP_v'}{}^0T(\theta_1, \dots, \theta_6) {}^{IP_v'}T^{-1} \quad (3.59)$$

3.3.2 Директни кинематички проблем елементарних хелија, односно емулираних 5-осних машина алатки

Као што је познато [71-73], позиције и оријентације координатног система алата $\{T\}$, у односу на референтни координатни систем $\{M\}$, су описане матрицама ${}^M_T T$, једначина (3.9), које се одређују за све типове емулираних 5-осних машина алатки, слике 3.1 до 3.7. Ове матрице ${}^M_T T$, које се као матрице бројних вредности добијају из *CL* датотеке, се у функцији унутрашњих координата могу изразити преко одговарајућих матрица ${}^M_0 T$, ${}^{IP}_T T$, ${}^{IP_v} T$, ${}^{IP_h} T$ и ${}^{IP_v'} T$ датих на сликама 3.1 до 3.7 и одговарајућег модела робота: 0T из једначине (3.27), ${}^{IP_v}{}^0T$ из једначине (3.34), ${}^{IP_h}{}^0T$ из једначине (3.42) или ${}^{IP_v'}{}^0T$ из једначине (3.50). Овако изражене матрице ${}^M_T T$, представљају решене директне кинематичке проблеме емулираних 5-осних машина алатки, при чему се из матрица оријентација ${}^M_T R$, односно

јединичних вектора оса алата ${}^M\mathbf{k}_T$, одређују одговарајући углови оријентације алата. Даље се дају процедуре за разматране типове:

- за емулиране 5-осне вертикалне глодалице, конфигурација к1.1, слика 3.1, и к3.1, слика 3.3, се применом датих хомогених трансформационих матрица ${}^M_0\mathbf{T}$, ${}^{IP}_T\mathbf{T}$, ${}^{IP}_v\mathbf{T}$ и једначине (3.41), добија

$${}^M_T\mathbf{T} = {}^M_0\mathbf{T} {}^0_{IP_v}\mathbf{T}(\theta_1, \dots, \theta_5) {}^{IP}_v\mathbf{T} = {}^M_0\mathbf{T} {}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_5, \theta_6 = 0^\circ(\text{блокирана})) {}^{IP}_T\mathbf{T} =$$

$$\begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.60)$$

На пример, заменом матрице ${}^{IP}_v\mathbf{T}(\theta_1, \dots, \theta_5)$ из једначине (3.34), чији су чланови дати једначинама (3.37) до (3.40), у једначину (3.60), добија се матрица ${}^M_T\mathbf{T}$ у функцији унутрашњих координата робота као

$${}^M_T\mathbf{T} = \begin{bmatrix} {}^M_T\mathbf{R} & {}^M\mathbf{p}_T \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} o_{IP_vy} & -n_{IP_vy} & a_{IP_vy} & -a_{IP_vy}l_T + p_{IP_vy} \\ -o_{IP_vx} & n_{IP_vx} & -a_{IP_vx} & a_{IP_vx}l_T - p_{IP_vx} \\ o_{IP_vz} & -n_{IP_vz} & a_{IP_vz} & -a_{IP_vz}l_T + p_{IP_vz} \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.61)$$

Исти резултат би се добио када би се у једначину (3.60) уврстила матрица ${}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_5, \theta_6 = 0^\circ(\text{блокирана}))$.

У једначини (3.61), вектор положаја врха алата у референтном координатним систему $\{M\}$

$${}^M\mathbf{p}_T = \begin{bmatrix} x_M \\ y_M \\ z_M \end{bmatrix} = \begin{bmatrix} -a_{IP_vy}l_T + p_{IP_vy} \\ a_{IP_vx}l_T - p_{IP_vx} \\ -a_{IP_vz}l_T + p_{IP_vz} \end{bmatrix} \quad (3.62)$$

и матрица оријентације алата ${}^M_T\mathbf{R}$ су изражени у функцији унутрашњих координата робота. Како емулирана машина алатка има 5 степени слободе, само је правац осе алата z_T управљив. То значи да се према једначини (3.61), из вектора ${}^M\mathbf{k}_T$, тј.

$${}^M\mathbf{k}_T = \begin{bmatrix} k_{Tx} \\ k_{Ty} \\ k_{Tz} \end{bmatrix} = \begin{bmatrix} a_{IP_vy} \\ -a_{IP_vx} \\ a_{IP_vz} \end{bmatrix}, \quad (3.63)$$

углови A и B могу израчунати, користећи једначине (3.12) и (3.13), као

$$A = \text{Atan2}\left(-k_{Ty}, \sqrt{1 - k_{Ty}^2}\right) = \text{Atan2}\left(a_{IP_{vx}}, \sqrt{1 - a_{IP_{vx}}^2}\right) \quad (3.64)$$

$$B = \text{Atan2}\left(\frac{k_{Tx}}{cA}, \frac{k_{Tz}}{cA}\right) = \text{Atan2}\left(\frac{a_{IP_{vy}}}{cA}, \frac{a_{IP_{vz}}}{cA}\right) \quad (3.65)$$

Овим је комплетиран вектор спољашњих координата, односно решен је директан кинематички проблем емулираних 5-осних вертикалних глодалица;

- за емулиране 5-осне хоризонталне глодалице, конфигурација к2.1, слика 3.2, и к3.2, слика 3.4, се применом одговарајућих хомогених трансформационих матрица ${}^M_0\mathbf{T}$, ${}^{IP}_T\mathbf{T}$, ${}^{IP}_h\mathbf{T}$ и једначина (3.27) и (3.42), добијају следеће једнакости

$${}^M_T\mathbf{T} = {}^M_0\mathbf{T} {}^{IP}_h\mathbf{T}(\theta_1, \dots, \theta_5) {}^{IP}_T\mathbf{T} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.66)$$

или

$${}^M_0\mathbf{T} {}^{IP}_T\mathbf{T}(\theta_1, \dots, \theta_4 = 0^\circ(\text{блокирана}), \theta_5, \theta_6) {}^{IP}_h\mathbf{T} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.67)$$

На пример заменом матрице ${}^{IP}_h\mathbf{T}(\theta_1, \dots, \theta_5)$, из једначине (3.42), чији су чланови дати једначинама (3.45) до (3.48), у једначину (3.66), добија се матрица ${}^M_T\mathbf{T}$ у функцији унутрашњих координата као

$${}^M_T\mathbf{T} = \begin{bmatrix} {}^M_T\mathbf{R} & {}^M_T\mathbf{p}_T \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} o_{IP_{hy}} & -a_{IP_{hy}} & n_{IP_{hy}} & a_{IP_{hy}}b_T - n_{IP_{hy}}l_T - p_{IP_{hy}} \\ -o_{IP_{hz}} & a_{IP_{hz}} & -n_{IP_{hz}} & -a_{IP_{hz}}b_T + n_{IP_{hz}}l_T + p_{IP_{hz}} \\ o_{IP_{hx}} & -a_{IP_{hx}} & n_{IP_{hx}} & a_{IP_{hx}}b_T - n_{IP_{hx}}l_T - p_{IP_{hx}} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.68)$$

Исти резултат би се добио када би се у једначину (3.67) уврстила матрица ${}^{IP}_T\mathbf{T}(\theta_1, \dots, \theta_4 = 0^\circ(\text{блокирана}), \theta_5, \theta_6)$.

У једначини (3.68) вектор положаја врха алата у референтном координатним систему $\{M\}$

$${}^M \mathbf{p}_T = \begin{bmatrix} x_M \\ y_M \\ z_M \end{bmatrix} = \begin{bmatrix} a_{IP_{hy}} b_T - n_{IP_{hy}} l_T - p_{IP_{hy}} \\ -a_{IP_{hz}} b_T + n_{IP_{hz}} l_T + p_{IP_{hz}} \\ a_{IP_{hx}} b_T - n_{IP_{hx}} l_T - p_{IP_{hx}} \end{bmatrix} \quad (3.69)$$

и матрица оријентације алата ${}^M \mathbf{R}$ су изражени у функцији унутрашњих координата робота. Како емулирана машина алатка има 5 степени слободе, само је правац осе алата z_T управљив. То значи да се према једначини (3.68), из вектора ${}^M \mathbf{k}_T$, тј.

$${}^M \mathbf{k}_T = \begin{bmatrix} k_{Tx} \\ k_{Ty} \\ k_{Tz} \end{bmatrix} = \begin{bmatrix} n_{IP_{hy}} \\ -n_{IP_{hz}} \\ n_{IP_{hx}} \end{bmatrix}, \quad (3.70)$$

углови A и B могу израчунати, користећи једначине (3.16) и (3.17), као

$$B = \text{Atan2} \left(k_{Tx}, \sqrt{1 - k_{Tx}^2} \right) = \text{Atan2} \left(n_{IP_{hy}}, \sqrt{1 - n_{IP_{hy}}^2} \right) \quad (3.71)$$

$$A = \text{Atan2} \left(\frac{-k_{Ty}}{c_B}, \frac{k_{Tz}}{c_B} \right) = \text{Atan2} \left(\frac{n_{IP_{hz}}}{c_B}, \frac{n_{IP_{hx}}}{c_B} \right) \quad (3.72)$$

Овим је комплетиран вектор спољашњих координата, односно решен је директан кинематички проблем емулираних 5-осних хоризонталних глодалица;

- за емулирану 5-осну вертикалну глодалицу, конфигурације к3.5, слика 3.7, се применом датих хомогених трансформационих матрица ${}^M_0 \mathbf{T}$, ${}^{IP_v'}_T \mathbf{T}$ у једначини (3.50), на сличан начин добија матрица ${}^M_T \mathbf{T}$ као

$${}^M_T \mathbf{T} = {}^M_0 \mathbf{T} {}^{IP_v'}_T \mathbf{T}(\theta_1, \dots, \theta_4 = \pm 90^\circ (\text{блокирана}), \theta_5, \theta_6) {}^{IP_v'}_T \mathbf{T} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.73)$$

Заменом матрице ${}^{IP_v'}_T \mathbf{T}(\theta_1, \dots, \theta_4 = \pm 90^\circ (\text{блокирана}), \theta_5, \theta_6)$ из једначине (3.50), чији су чланови дати једначинама (3.53) до (3.56), у једначину (3.73) добија се матрица ${}^M_T \mathbf{T}$ у функцији унутрашњих координата робота као

$${}^M_T\mathbf{T} = \begin{bmatrix} {}^M_T\mathbf{R} & {}^M\mathbf{p}_T \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} o_{IP_v'y} & -n_{IP_v'y} & a_{IP_v'y} & -a_{IP_v'y}l_T + p_{IP_v'y} \\ -o_{IP_v'x} & n_{IP_v'x} & -a_{IP_v'x} & a_{IP_v'x}l_T - p_{IP_v'x} \\ o_{IP_v'z} & -n_{IP_v'z} & a_{IP_v'z} & -a_{IP_v'z}l_T + p_{IP_v'z} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.74)$$

У једначини (3.74), вектор положаја врха алата у референтном координатним систему $\{M\}$

$${}^M\mathbf{p}_T = \begin{bmatrix} x_M \\ y_M \\ z_M \end{bmatrix} = \begin{bmatrix} -a_{IP_v'y}l_T + p_{IP_v'y} \\ a_{IP_v'x}l_T - p_{IP_v'x} \\ -a_{IP_v'z}l_T + p_{IP_v'z} \end{bmatrix} \quad (3.75)$$

и матрица оријентације алата ${}^M_T\mathbf{R}$ су изражени у функцији унутрашњих координата робота. Како емулирана машина алатка има 5 степени слободе, само је правац осе алата z_T управљив. То значи да се из једначине (3.74), из вектора ${}^M\mathbf{k}_T$, тј.

$${}^M\mathbf{k}_T = \begin{bmatrix} k_{Tx} \\ k_{Ty} \\ k_{Tz} \end{bmatrix} = \begin{bmatrix} a_{IP_v'y} \\ -a_{IP_v'x} \\ a_{IP_v'z} \end{bmatrix}, \quad (3.76)$$

углови A и C могу израчунати, користећи једначине (3.21) и (3.22), као

$$A = \text{Atan2}\left(\sqrt{1 - k_{Tz}^2}, k_{Tz}\right) = \text{Atan2}\left(\sqrt{1 - a_{IP_v'y}^2}, a_{IP_v'y}\right) \quad (3.77)$$

$$C = \text{Atan2}\left(\frac{k_{Tx}}{sA}, \frac{-k_{Ty}}{sA}\right) = \text{Atan2}\left(\frac{a_{IP_v'y}}{sA}, \frac{a_{IP_v'x}}{sA}\right) \quad (3.78)$$

Овим је комплетиран вектор спољашњих координата, односно решен је директан кинематички проблем емулиране 5-осне вертикалне глодалице;

- за 5-осну вертикалну и хоризонталну глодалицу емулирану роботом са 6 степени слободе, код кога су све осе активне, конфигурације к3.3, слика 3.5, и к3.4, слика 3.6, се применом одговарајућих хомогених трансформационих матрица ${}^M_0\mathbf{T}$, ${}^{IP}_T\mathbf{T}$ са слике 3.5, односно слике 3.6 добија матрица ${}^M_T\mathbf{T}$ као

$${}^M_T\mathbf{T} = {}^M_0\mathbf{T} {}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_6) {}^{IP}_T\mathbf{T} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.79)$$

За емулирану 5-осну вертикалну глодалицу се заменом матрице ${}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_6)$ из једначине (3.27), чији су чланови дати једначинама (3.30) до (3.33), у једначину (3.79), добија матрица ${}^M_T\mathbf{T}$ у функцији унутрашњих координата робота као

$${}^M_T\mathbf{T} = \begin{bmatrix} {}^M_T\mathbf{R} & {}^M\mathbf{p}_T \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} o_{IPy} & -n_{IPy} & a_{IPy} & -a_{IPy}l_T + p_{IPy} \\ -o_{IPx} & n_{IPx} & -a_{IPx} & a_{IPx}l_T - p_{IPx} \\ o_{IPz} & -n_{IPz} & a_{IPz} & -a_{IPz}l_T + p_{IPz} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.80)$$

док се за емулирану 5-осну хоризонталну глодалицу матрица ${}^M_T\mathbf{T}$ добија као

$${}^M_T\mathbf{T} = \begin{bmatrix} {}^M_T\mathbf{R} & {}^M\mathbf{p}_T \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} o_{IPy} & -a_{IPy} & n_{IPy} & a_{IPy}b_T - n_{IPy}l_T - p_{IPy} \\ -o_{IPz} & a_{IPz} & -n_{IPz} & -a_{IPz}b_T + n_{IPz}l_T + p_{IPz} \\ o_{IPx} & -a_{IPx} & n_{IPx} & a_{IPx}b_T - n_{IPx}l_T - p_{IPx} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} i_{Tx} & j_{Tx} & k_{Tx} & x_M \\ i_{Ty} & j_{Ty} & k_{Ty} & y_M \\ i_{Tz} & j_{Tz} & k_{Tz} & z_M \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.81)$$

У једначинама (3.80) и (3.81), вектори положаја врха алата у референтним координатним системима $\{M\}$ емулиране 5-осне вертикалне глодалице

$${}^M\mathbf{p}_T = \begin{bmatrix} x_M \\ y_M \\ z_M \end{bmatrix} = \begin{bmatrix} -a_{IPy}l_T + p_{IPy} \\ a_{IPx}l_T - p_{IPx} \\ -a_{IPz}l_T + p_{IPz} \end{bmatrix}, \quad (3.82)$$

односно емулиране 5-осне хоризонталне глодалице

$${}^M\mathbf{p}_T = \begin{bmatrix} x_M \\ y_M \\ z_M \end{bmatrix} = \begin{bmatrix} a_{IPy}b_T - n_{IPy}l_T - p_{IPy} \\ -a_{IPz}b_T + n_{IPz}l_T + p_{IPz} \\ a_{IPx}b_T - n_{IPx}l_T - p_{IPx} \end{bmatrix} \quad (3.83)$$

и одговарајуће матрице оријентација ${}^M_T\mathbf{R}$ су изражени у функцији унутрашњих координата. Како емулирана машина алатка има 5 степени слободе, само је правац осе алата z_T управљив. То значи да се за емулирану 5-осну вертикалну глодалицу, једначина (3.80), из вектора

$${}^M\mathbf{k}_T = \begin{bmatrix} k_{Tx} \\ k_{Ty} \\ k_{Tz} \end{bmatrix} = \begin{bmatrix} a_{IPy} \\ -a_{IPx} \\ a_{IPz} \end{bmatrix} \quad (3.84)$$

углови A и B се могу израчунати, користећи једначине (3.12) и (3.13), као

$$A = \text{Atan2}\left(-k_{Ty}, \sqrt{1 - k_{Ty}^2}\right) = \text{Atan2}\left(a_{IPvx}, \sqrt{1 - a_{IPvx}^2}\right) \quad (3.85)$$

$$B = \text{Atan2}\left(\frac{k_{Tx}}{cA}, \frac{k_{Tz}}{cA}\right) = \text{Atan2}\left(\frac{a_{IPvy}}{cA}, \frac{a_{IPvz}}{cA}\right) \quad (3.86)$$

За емулирану 5-осну хоризонталну глодалицу, једначина (3.81), из вектора

$${}^M\mathbf{k}_T = \begin{bmatrix} k_{Tx} \\ k_{Ty} \\ k_{Tz} \end{bmatrix} = \begin{bmatrix} n_{IPy} \\ -n_{IPz} \\ n_{IPx} \end{bmatrix} \quad (3.87)$$

углови A и B се могу израчунати, користећи једначине (3.16) и (3.17), као

$$B = \text{Atan2}\left(k_{Tx}, \sqrt{1 - k_{Tx}^2}\right) = \text{Atan2}\left(n_{IPhy}, \sqrt{1 - n_{IPhy}^2}\right) \quad (3.88)$$

$$A = \text{Atan2}\left(\frac{-k_{Ty}}{cB}, \frac{k_{Tz}}{cB}\right) = \text{Atan2}\left(\frac{n_{IPhz}}{cB}, \frac{n_{IPhx}}{cB}\right) \quad (3.89)$$

Овим су комплетирани вектори спољашњих координата, односно решени су директни кинематички проблеми за дате емулиране глодалице.

3.4 Инверзни кинематички проблем робота са 5 и 6 степени слободe конфигурисаних за емулирање 5-осних машина алатки

У циљу примене генерализованог приступа у решавању инверзног кинематичког проблема робота са 5 и 6 степени слободe, најпре је потребно на основу вектора спољашњих координата, претходно анализираних емулираних 5-осних глодалица, односно вектора положаја врха алата $[x_M \ y_M \ z_M]^T$ и одговарајућих углова оријентације (A, B) или (A, C) , одредити положаје и оријентације одговарајућих координатних система додатних интерфејса $\{IP\}$ у односу на базе координатне системе робота $\{0\}$, односно $\{{}^0\mathbf{p}_{0IP}, {}^0\mathbf{R}\}$, $\{{}^0\mathbf{p}_{0IP_v}, {}^0\mathbf{R}\}$, $\{{}^0\mathbf{p}_{0IP_v'}, {}^0\mathbf{R}\}$ и $\{{}^0\mathbf{p}_{0IP_h}, {}^0\mathbf{R}\}$. На овај начин су практично директни кинематички проблеми елементарних обрадних ћелија на бази робота, односно емулираних 5-осних машина алатки сведени на директне кинематичке проблеме робота са 5 и 6 степени слободe из којих могу решити њихови инверзни кинематички проблеми. Овај поступак се даље уопштено показује за све претходно разматране случајеве:

- за 5-осну вертикалну глодалицу у конфигурацијама к1.1, слика 3.1, и к3.1, слика 3.3, из једначине (3.60) се добијају једначине

$${}_{IP_v}^0\mathbf{T}(\theta_1, \dots, \theta_5) = {}_0^M\mathbf{T}^{-1} {}_T^M\mathbf{T}^{IP_v} \mathbf{T}^{-1} \quad (3.90)$$

и

$${}_{IP}^0\mathbf{T}(\theta_1, \dots, \theta_5, \theta_6 = 0^\circ \text{(блокирана)}) = {}_0^M\mathbf{T}^{-1} {}_T^M\mathbf{T}^{IP} \mathbf{T}^{-1} \quad (3.91)$$

Даље се заменом ${}_{IP_v}^0\mathbf{T}(\theta_1, \dots, \theta_5)$ из једначине (3.37) у једначину (3.90) и заменом ${}_{IP}^0\mathbf{T}(\theta_1, \dots, \theta_5, \theta_6 = 0^\circ \text{(блокирана)})$ из једначине (3.27) у једначину (3.91), добијају следеће једначине

$${}_{IP_v}^0\mathbf{T}(\theta_1, \dots, \theta_5) = \begin{bmatrix} {}_{IP_v}^0\mathbf{R}(\theta_1, \dots, \theta_5) & {}^0\mathbf{p}_{0IP_v}(\theta_1, \dots, \theta_5) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_{IP_vx} & o_{IP_vx} & a_{IP_vx} & p_{IP_vx} \\ n_{IP_vy} & o_{IP_vy} & a_{IP_vy} & p_{IP_vy} \\ n_{IP_vz} & o_{IP_vz} & a_{IP_vz} & p_{IP_vz} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} j_{Ty} & -i_{Ty} & -k_{Ty} & -y_M - k_{Ty}l_T \\ -j_{Tx} & i_{Tx} & k_{Tx} & x_M + k_{Tx}l_T \\ -j_{Tz} & i_{Tz} & k_{Tz} & z_M + k_{Tz}l_T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.92)$$

и

$$\begin{aligned}
 & {}_{IP}^0\mathbf{T}(\theta_1, \dots, \theta_5, \theta_6 = 0^\circ (\text{блокирана})) = \\
 & \begin{bmatrix} {}_{IP}^0\mathbf{R}(\theta_1, \dots, \theta_5, \theta_6 = 0^\circ) & {}^0\mathbf{p}_{0IP}(\theta_1, \dots, \theta_5, \theta_6 = 0^\circ) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\
 & \begin{bmatrix} n_{IPx} & o_{IPx} & a_{IPx} & p_{IPx} \\ n_{IPy} & o_{IPy} & a_{IPy} & p_{IPy} \\ n_{IPz} & o_{IPz} & a_{IPz} & p_{IPz} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} j_{Ty} & -i_{Ty} & -k_{Ty} & -y_M - k_{Ty}l_T \\ -j_{Tx} & i_{Tx} & k_{Tx} & x_M + k_{Tx}l_T \\ -j_{Tz} & i_{Tz} & k_{Tz} & z_M + k_{Tz}l_T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.93)
 \end{aligned}$$

Како се за добијену позицију и оријентацију врха алата, односно вектор спољашњих координата $\mathbf{x} = [x_M \ y_M \ z_M \ A \ B]^T$, из програма обраде (*G*-кода), формира матрица ${}^M_T\mathbf{T}$ као матрица бројних вредности, то се из једначина (3.90) и (3.91), односно (3.92) и (3.93), чије су леве стране функције унутрашњих координата, једначине (3.37) до (3.40), односно једначине (3.30) до (3.33), решава инверзни кинематички проблем;

- за 5-осну хоризонталну глодалицу у конфигурацијама к2.1, слика 3.2, и к3.2, слика 3.4 из једначине (3.66) се добијају једначине

$${}_{IP_h}^0\mathbf{T}(\theta_1, \dots, \theta_5) = {}^M_0\mathbf{T}^{-1} {}^M_T\mathbf{T} {}^{IP_h}_T\mathbf{T}^{-1} \quad (3.94)$$

и

$${}_{IP}^0\mathbf{T}(\theta_1, \dots, \theta_4 = 0^\circ (\text{блокирана}), \theta_5, \theta_6) = {}^M_0\mathbf{T}^{-1} {}^M_T\mathbf{T} {}^{IP}_T\mathbf{T}^{-1} \quad (3.95)$$

Даље се заменом ${}_{IP_h}^0\mathbf{T}(\theta_1, \dots, \theta_5)$ из једначине (3.42) у једначину (3.94) и заменом ${}_{IP}^0\mathbf{T}(\theta_1, \dots, \theta_4 = 0^\circ (\text{блокирана}), \theta_5, \theta_6)$ из једначине (3.27) у једначину (3.95), добијају следеће једначине

$$\begin{aligned}
 & {}_{IP_h}^0\mathbf{T}(\theta_1, \dots, \theta_5) = \begin{bmatrix} {}_{IP_h}^0\mathbf{R}(\theta_1, \dots, \theta_5) & {}^0\mathbf{p}_{0IP_h}(\theta_1, \dots, \theta_5) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\
 & \begin{bmatrix} n_{IP_hx} & o_{IP_hx} & a_{IP_hx} & p_{IP_hx} \\ n_{IP_hy} & o_{IP_hy} & a_{IP_hy} & p_{IP_hy} \\ n_{IP_hz} & o_{IP_hz} & a_{IP_hz} & p_{IP_hz} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\
 & \begin{bmatrix} k_{Tz} & i_{Tz} & -j_{Tz} & -z_M - k_{Tz}l_T - j_{Tz}b_T \\ k_{Tx} & i_{Tx} & -j_{Tx} & -x_M - k_{Tx}l_T - j_{Tx}b_T \\ -k_{Ty} & -i_{Ty} & j_{Ty} & y_M + k_{Ty}l_T + j_{Ty}b_T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.96)
 \end{aligned}$$

и

$$\begin{aligned}
 & {}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_4 = 0^\circ (\text{блокирана}), \theta_5, \theta_6) = \\
 & \left[\begin{array}{ccc|c}
 {}^0_{IP}\mathbf{R}(\theta_1, \dots, \theta_4 = 0^\circ, \theta_5, \theta_6) & & & {}^0\mathbf{p}_{0IP}(\theta_1, \dots, \theta_4 = 0^\circ, \theta_5, \theta_6) \\
 0 & 0 & 0 & 1
 \end{array} \right] = \\
 & \left[\begin{array}{cccc}
 n_{IPx} & o_{IPx} & a_{IPx} & p_{IPx} \\
 n_{IPy} & o_{IPy} & a_{IPy} & p_{IPy} \\
 n_{IPz} & o_{IPz} & a_{IPz} & p_{IPz} \\
 0 & 0 & 0 & 1
 \end{array} \right] = \\
 & \left[\begin{array}{cccc}
 k_{Tz} & i_{Tz} & -j_{Tz} & -z_M - k_{Tz}l_T - j_{Tz}b_T \\
 k_{Tx} & i_{Tx} & -j_{Tx} & -x_M - k_{Tx}l_T - j_{Tx}b_T \\
 -k_{Ty} & -i_{Ty} & j_{Ty} & y_M + k_{Ty}l_T + j_{Ty}b_T \\
 0 & 0 & 0 & 1
 \end{array} \right] \quad (3.97)
 \end{aligned}$$

Како се за добијену позицију и оријентацију врха алата, односно вектор спољашњих координата $\mathbf{x} = [x_M \ y_M \ z_M \ A \ B]^T$, из програма обраде (*G*-кôда), формира матрица ${}^M_T\mathbf{T}$ као матрица бројних вредности, то се из једначине (3.94) и (3.95), односно (3.96) и (3.97), чије су леве стране функције унутрашњих координата, једначине (3.45) до (3.48), односно (3.30) до (3.33), решава инверзни кинематички проблем;

- за 5-осну вертикалну глодалицу у конфигурацији к3.5, слика 3.7, једначина (3.73), добија се једначина

$${}^0_{IP_v}\mathbf{T}(\theta_1, \dots, \theta_4 = \pm 90^\circ (\text{блокирана}), \theta_5, \theta_6) = {}^M_0\mathbf{T}^{-1} {}^M_T \mathbf{T}^{IP_v'} \mathbf{T}^{-1} \quad (3.98)$$

Даље се заменом ${}^0_{IP_v}\mathbf{T}(\theta_1, \dots, \theta_4 = \pm 90^\circ (\text{блокирана}), \theta_5, \theta_6)$ из једначине (3.50) у једначину (3.98), добија следећа једначина

$$\begin{aligned}
 & {}^0_{IP_v}\mathbf{T}(\theta_1, \dots, \theta_4 = \pm 90^\circ (\text{блокирана}), \theta_5, \theta_6) = \\
 & \left[\begin{array}{ccc|c}
 {}^0_{IP_v}\mathbf{R}(\theta_1, \dots, \theta_4 = \pm 90^\circ, \theta_5, \theta_6) & & & {}^0\mathbf{p}_{0IP_v'}(\theta_1, \dots, \theta_4 = \pm 90^\circ, \theta_5, \theta_6) \\
 0 & 0 & 0 & 1
 \end{array} \right] = \\
 & \left[\begin{array}{cccc}
 j_{Ty} & -i_{Ty} & -k_{Ty} & -y_M - k_{Ty}l_T \\
 -j_{Tx} & i_{Tx} & k_{Tx} & x_M + k_{Tx}l_T \\
 -j_{Tz} & i_{Tz} & k_{Tz} & z_M + k_{Tz}l_T \\
 0 & 0 & 0 & 1
 \end{array} \right] \quad (3.99)
 \end{aligned}$$

Како се за добијену позицију и оријентацију врха алата, односно вектор спољашњих координата $\mathbf{x} = [x_M \ y_M \ z_M \ A \ B]^T$, из програма обраде (*G*-кôда), формира матрица ${}^M_T\mathbf{T}$ као матрица бројних вредности, то се из једначине (3.98), односно (3.99), чије су леве стране функције

унутрашњих координата, једначине (3.53) до (3.56), решава инверзни кинематички проблем;

- за 6-осну вертикалну глодалицу коју емулира робот са 6 степени слободе у конфигурацији к3.3, слика 3.5, при чему су све осе робота активне, једначина (3.79), добија се једначина

$${}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_6) = {}^M_0\mathbf{T}^{-1} {}^M_T\mathbf{T} {}^{IP}_T\mathbf{T}^{-1} \quad (3.100)$$

Даље се заменом ${}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_6)$ из једначине (3.27) у једначину (3.100), добија следећа једначина

$${}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_6) = \begin{bmatrix} {}^0_{IP}\mathbf{R}(\theta_1, \dots, \theta_6) & {}^0\mathbf{p}_{0IP}(\theta_1, \dots, \theta_6) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} j_{Ty} & -i_{Ty} & -k_{Ty} & -y_M - k_{Ty}l_T \\ -j_{Tx} & i_{Tx} & k_{Tx} & x_M + k_{Tx}l_T \\ -j_{Tz} & i_{Tz} & k_{Tz} & z_M + k_{Tz}l_T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.101)$$

Како се за добијену позицију и оријентацију врха алата, односно вектор спољашњих координата $\mathbf{x} = [x_M \ y_M \ z_M \ A \ B]^T$, из програма обраде (G-кода), формира матрица ${}^M_T\mathbf{T}$ као матрица бројних вредности, то се из једначине (3.100) односно (3.101), чије су леве стране функције унутрашњих координата, једначине (3.30) до (3.33), решава инверзни кинематички проблем;

- за 6-осну хоризонталну глодалицу која је емулирана роботом са 6 степени слободе у конфигурацији к3.4, слика 3.6, при чему су све осе робота активне, једначина (3.79), добија се једначина:

$${}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_6) = {}^M_0\mathbf{T}^{-1} {}^M_T\mathbf{T} {}^{IP}_T\mathbf{T}^{-1} \quad (3.102)$$

Даље се заменом ${}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_6)$ из једначине (3.27) у једначину (3.102), добија следећа једначина

$${}^0_{IP}\mathbf{T}(\theta_1, \dots, \theta_6) = \begin{bmatrix} {}^0_{IP}\mathbf{R}(\theta_1, \dots, \theta_6) & {}^0\mathbf{p}_{0IP}(\theta_1, \dots, \theta_6) \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} k_{Tz} & i_{Tz} & -j_{Tz} & -z_M - k_{Tz}l_T - j_{Tz}b_T \\ k_{Tx} & i_{Tx} & -j_{Tx} & -x_M - k_{Tx}l_T - j_{Tx}b_T \\ -k_{Ty} & -i_{Ty} & j_{Ty} & y_M + k_{Ty}l_T + j_{Ty}b_T \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.103)$$

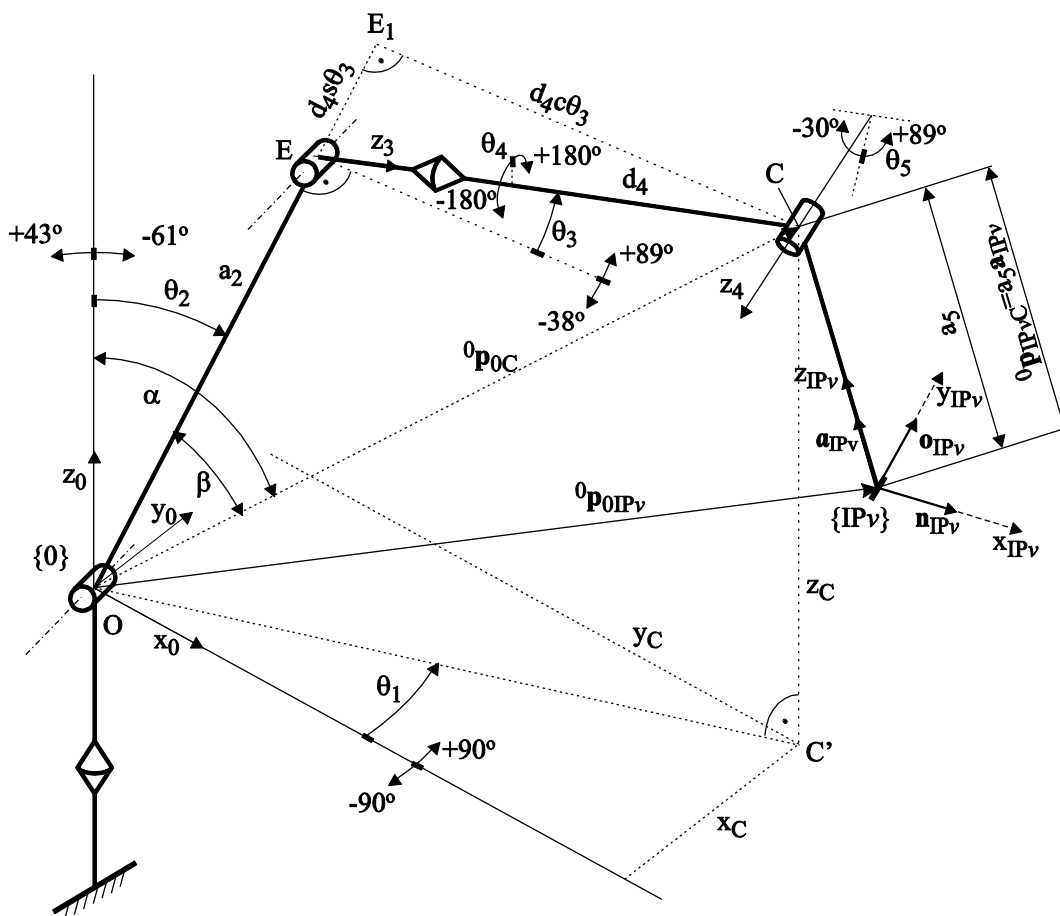
Како се за добијену позицију и оријентацију врха алата, односно вектор спољашњих координата $\mathbf{x} = [x_M \ y_M \ z_M \ A \ B]^T$, из програма обраде (*G*-кода), формира матрица ${}^M_T\mathbf{T}$ као матрица бројних вредности, то се из једначине (3.102) односно (3.103), чије су леве стране функције унутрашњих координата, једначине (3.30) до (3.33), решава инверзни кинематички проблем.

За решавање инверзног кинематичког проблема разматраних емулираних ћелија за обраду на бази робота вертикалне зглобне конфигурације, слике 3.1 до 3.7, коришћен је *Pieper*-ов метод. *Pieper*-ов метод, чији је детаљан опис дат у [71], који се користи за манипулаторе чије се три последње осе секу у једној тачки или чије су три суседне осе паралелне. Овај метод омогућава раздвајање решавања инверзног кинематичког проблема на инверзни кинематички проблем позиције, који се обично решава геометријски, и инверзни кинематички проблем оријентације, који се обично решава аналитички. У литератури постоји неколико примера решавања инверзног кинематичког проблема робота са пет степени слободе. У [75] и [76] је дата могућност примене ове методе за решавања инверзног кинематичког проблема робота са пет ротационих степени слободе кретања. При томе су за решавање инверзних кинематичких проблема овде разматраних робота, који емулирају 5-осне машине алатке, у матрицама оријентација од интереса само колоне у којима фигурише јединични вектор осе алата \mathbf{k}_T , једначине (3.92), (3.96), (3.99), (3.101) и (3.103). Даље се, као пример, даје комплетно решење инверзног кинематичког проблема, применом *Pieper*-овог метода, специјализованог робота са 5 степени слободе, који емулира 5-осну вертикалну глодалицу конфигурације (X, Y, Z, A, B).

3.4.1 Пример комплетног решења инверзног кинематичког проблема специјализованог робота вертикалне зглобне конфигурације са 5 степени слободе који емулира 5-оосну вертикалну глодалицу

У случају специјализованог робота са 5 степени слободе, слика 3.1, осе оријентације алата, односно осе последња два зглоба z_3 и z_4 , се секу у тачки C , слика 3.10. За решавање инверзног кинематичког проблема у овој дисертацији, коришћени су делом геометријски и алгебарски приступ.

Као што се може видети са слике 3.13, позицију тачке C одређују унутрашње координате θ_1, θ_2 и θ_3 .



Слика 3.13 Кинематичко декупловање

За задати вектор спољашњих координата емулираних 5-осних машина алатки $x = [x_M \ y_M \ z_M \ A \ B]^T$ најпре је према једначини (3.92) одређен вектор положаја ${}^0p_{0IP_v}$ као

$${}^0p_{0IP_v} = \begin{bmatrix} p_{IP_v x} \\ p_{IP_v y} \\ p_{IP_v z} \end{bmatrix} = \begin{bmatrix} -y_M - k_{Ty} l_T \\ x_M + k_{Tx} l_T \\ z_M + k_{Tz} l_T \end{bmatrix}, \quad (3.104)$$

а затим према једначинама (3.10) и (3.92) и матрица ротације ${}_{IP_v}^0R$ као

$${}_{IP_v}^0R = \begin{bmatrix} n_{IP_v x} & o_{IP_v x} & a_{IP_v x} \\ n_{IP_v y} & o_{IP_v y} & a_{IP_v y} \\ n_{IP_v z} & o_{IP_v z} & a_{IP_v z} \end{bmatrix} = \begin{bmatrix} j_{Ty} & -i_{Ty} & -k_{Ty} \\ -j_{Tx} & i_{Tx} & k_{Tx} \\ -j_{Tz} & i_{Tz} & k_{Tz} \end{bmatrix} = \begin{bmatrix} cB & sAsB & cAsB \\ 0 & cA & -sA \\ -sB & sAcB & cAcB \end{bmatrix} \quad (3.105)$$

што одређује позицију и оријентацију координатног система додатног интерфејса IP_v . Овим је, као што је раније наглашено, проблем решавања инверзног кинематичког проблема емулиране 5-осне машине алатке сведен на познати поступак решавања инверзног кинематичког проблема специјализованог робота са 5 степени слободе, датог на слици 3.13.

Како се при 5-осној обради управља само осом алата z_T , односно у овом случају осом интерфејса z_{IP_v} , то је у матрици ${}^{0}_{IP_v}R$ из једначине (3.105) од интереса само вектор $\mathbf{a}_{IP_v} = \mathbf{k}_T$. Сада се вектор положаја тачке C може израчунати као

$${}^0\mathbf{p}_{0C} = \begin{bmatrix} x_C \\ y_C \\ z_C \end{bmatrix} = {}^0\mathbf{p}_{0IP_v} + {}^0\mathbf{p}_{IP_vC} = {}^0\mathbf{p}_{0IP_v} + {}^{IP_v}{}^0R \cdot \begin{bmatrix} 0 \\ 0 \\ a_5 \end{bmatrix} = {}^0\mathbf{p}_{0IP_v} + \mathbf{a}_{IP_v} \cdot a_5 = \begin{bmatrix} p_{IP_vx} + a_5 a_{IP_vx} \\ p_{IP_vy} + a_5 a_{IP_vy} \\ p_{IP_vz} + a_5 a_{IP_vz} \end{bmatrix} \quad (3.106)$$

На основу израчунатих компонената вектора ${}^0\mathbf{p}_{0C}$, геометријским приступом је решен инверзни кинематички проблем позиције. Посматрајући пројекцију тачке C у равни $(X_M, -Y_M)$, слика 3.1, односно равни (x_0, y_0) , слика 3.13, и с обзиром на референтни положај робота, угао θ_1 се израчунава као

$$\theta_1 = \text{Atan2}(y_C, x_C) \quad (3.107)$$

Усвојене границе угла $-90^\circ \leq \theta_1 \leq 90^\circ$, односно чињеница да је аргумент $x_C \geq 0$ у функцији Atan2 , једнозначно одређују вредност угла θ_1 .

Узимајући у обзир пројекцију тачке C у равни коју формирају други и трећи сегмент робота, израчунати су углови θ_2 и θ_3 . Применом косинусне теореме на троугао OCE , $s\theta_3$ се израчунава као

$$s\theta_3 = \frac{x_C^2 + y_C^2 + z_C^2 - a_2^2 - d_4^2}{2 \cdot a_2 \cdot d_4} \quad (3.108)$$

Како су границе за угао θ_3 усвојене у опсегу $-30^\circ \leq \theta_3 \leq 89^\circ$, решење за θ_3 се добија као

$$\theta_3 = \text{Atan2}\left(s\theta_3, \sqrt{1 - s\theta_3^2}\right) \quad (3.109)$$

Коришћењем решења за угао θ_3 , које условљава решење за угао θ_2 , као и углове α и β који могу бити израчунати посматрањем троуглова $OC'C$ и OCE_1 , угао θ_2 се израчунава као

$$\theta_2 = \text{Atan2}\left(-\sqrt{x_c^2 - y_c^2}, x_z\right) + \text{Atan2}(d_4 \cdot c\theta_3, a_2 + d_4 \cdot s\theta_3) \quad (3.110)$$

при чему су узете у обзир усвојене границе $-61^\circ \leq \theta_2 \leq 43^\circ$, које одређују референтни положај угла θ_2 .

Геометријским приступом израчунате вредности углова θ_1 , θ_2 и θ_3 се даље користе за аналитичко решавање инверзног кинематичког проблема оријентације, односно одређивање углова θ_4 и θ_5 на следећи начин. Полазећи од једначине (3.34) која се може написати као

$${}_{IP_v}^0\mathbf{T} = {}_1^0\mathbf{A}_2^1\mathbf{A}_3^2\mathbf{A}_4^3\mathbf{A}_5^4\mathbf{A}_{IP_v}^5\mathbf{T} = {}_3^0\mathbf{T} \cdot {}_{IP_v}^3\mathbf{T} \quad (3.111)$$

где су

$${}_3^0\mathbf{T} = \begin{bmatrix} {}_3^0\mathbf{R} & {}^0\mathbf{p}_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}_1^0\mathbf{A}_2^1\mathbf{A}_3^2\mathbf{A} \quad (3.112)$$

$${}_{IP_v}^3\mathbf{T} = \begin{bmatrix} {}_3^0\mathbf{R} & {}^0\mathbf{p}_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}_4^3\mathbf{A}_5^4\mathbf{A}_{IP_v}^5\mathbf{T} \quad (3.113)$$

тако да се добија да је

$${}_{IP_v}^0\mathbf{R} = {}_3^0\mathbf{R} \cdot {}_{IP_v}^3\mathbf{R} \quad (3.114)$$

Заменом матрица ${}^{i-1}_i\mathbf{A}$, $i = 1, 2, 3$ са претходно добијеним угловима θ_1 , θ_2 и θ_3 , добија се

$${}_3^0\mathbf{R} = \begin{bmatrix} -c\theta_1 \cdot s\theta_{23} & s\theta_1 & c\theta_1 \cdot c\theta_{23} \\ -s\theta_1 \cdot s\theta_{23} & -c\theta_1 & s\theta_1 \cdot c\theta_{23} \\ c\theta_{23} & 0 & s\theta_{23} \end{bmatrix} \quad (3.115)$$

као матрица бројних вредности. Затим се заменом матрица ${}^{i-1}_i\mathbf{A}$, $i = 4, 5$ са непознатим вредностима углова θ_4 и θ_5 , добија да је

$${}_{IP_v}^3\mathbf{R} = \begin{bmatrix} * & * & c\theta_4 \cdot c\theta_5 \\ * & * & -s\theta_4 \cdot c\theta_5 \\ * & * & -s\theta_5 \end{bmatrix} \quad (3.116)$$

при чему су непотребни чланови за израчунавање углова оса оријентације θ_4 и θ_5 означени звездицама.

С обзиром да су у једначини (3.114) матрице ${}_{IP_v}^0\mathbf{R}$ и ${}_3^0\mathbf{R}$ познате, као матрице бројних вредности, то се матрица ${}_{IP_v}^3\mathbf{R}$, као матрица бројних вредности, може одредити као

$${}_{IP_v}^3\mathbf{R} = {}_3^0\mathbf{R}^{-1} \cdot {}_{IP_v}^0\mathbf{R} \quad (3.117)$$

Како је од интереса оријентација осе алата z_T , чији се правац поклапа са јединичним вектором \mathbf{a}_{IP_v} , једначина (3.105), из једначине (3.117) је очигледно да је:

$${}^3\mathbf{a}_{IP_v} = {}^0\mathbf{R}(\theta_4, \theta_5)^{-1} \cdot {}^0\mathbf{a}_{IP_v} \quad (3.118)$$

што се према једначинама (3.115) и (3.116) може изразити као

$$\begin{bmatrix} c\theta_4 \cdot c\theta_5 \\ -s\theta_4 \cdot c\theta_5 \\ -s\theta_5 \end{bmatrix} = \begin{bmatrix} -c\theta_1 \cdot s\theta_{23} \cdot a_{IP_vx} - s\theta_1 \cdot s\theta_{23} \cdot a_{IP_vy} + c\theta_{23} \cdot a_{IP_vz} \\ s\theta_1 \cdot a_{IP_vx} - c\theta_1 \cdot a_{IP_vy} \\ c\theta_1 \cdot c\theta_{23} \cdot a_{IP_vx} + s\theta_1 \cdot c\theta_{23} \cdot a_{IP_vy} + s\theta_{23} \cdot a_{IP_vz} \end{bmatrix} \quad (3.119)$$

Узимајући у обзир да је $-30^\circ \leq \theta_5 \leq 89^\circ$, из једначине (3.119) се угао θ_5 израчунава као:

$$\theta_5 = \text{Atan2}(s\theta_5, c\theta_5) \quad (3.120)$$

где је

$$s\theta_5 = -c\theta_1 \cdot c\theta_{23} \cdot a_{IP_vx} - s\theta_1 \cdot c\theta_{23} \cdot a_{IP_vy} - s\theta_{23} \cdot a_{IP_vz} \quad (3.121)$$

и

$$c\theta_5 = \sqrt{1 - s\theta_5^2} \quad (3.122)$$

Угао θ_4 се из једначине (3.119) може израчунати као

$$\theta_4 = \text{Atan2}(s\theta_4, c\theta_4) \quad (3.123)$$

где је

$$s\theta_4 = (-s\theta_1 \cdot a_{IP_vx} + c\theta_1 \cdot a_{IP_vy}) / c\theta_5 \quad (3.124)$$

и

$$c\theta_4 = (-c\theta_1 \cdot s\theta_{23} \cdot a_{IP_vx} - s\theta_1 \cdot s\theta_{23} \cdot a_{IP_vy} + c\theta_{23} \cdot a_{IP_vz}) / c\theta_5 \quad (3.125)$$

чиме је решење инверзног кинематичког проблема за дати пример комплетирано.

Применом приказаног комбинованог геометријско-аналитичког приступа, решени су сви остали разматрани случајеви емулираних хелија за обраду, на бази робота вертикалне зглобне конфигурације са слика 3.1 до 3.7.

3.5 Анализа радног простора елементарних ћелија на бази робота као емулираних 5-осних машина алатки

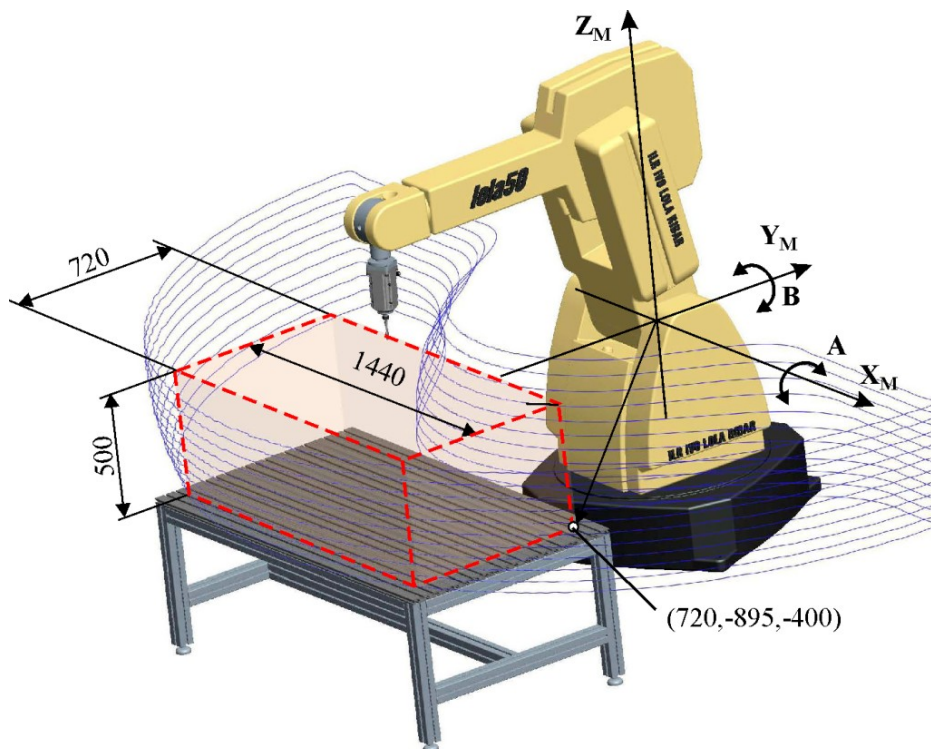
Коришћењем решења инверзног кинематичког проблема, могуће је одредити радне просторе позиција и оријентација елементарних ћелија за обраду на бази робота са 5 и 6 степени слободе као емулираних 5-осних машина алатки. Приступ који је овде примењен за одређивање радног простора је врло користан, без обзира што је у литератури [77] примењен за одрађивање радног простора позиције и оријентације машина са паралелном кинематиком. Као пример, овде се даје анализа радног простора елементарне ћелије за обраду на бази робота са 6 степени слободе који емулира 5-осну вертикалну глодалицу (X, Y, Z, A, B) , конфигурација к3.1 са слике 3.3. За овај случај се радни простор позиције и оријентације може изразити Буловом функцијом као

$$W_s(X_M, Y_M, Z_M, A_M, B_M) = \{0,1\} \quad (3.126)$$

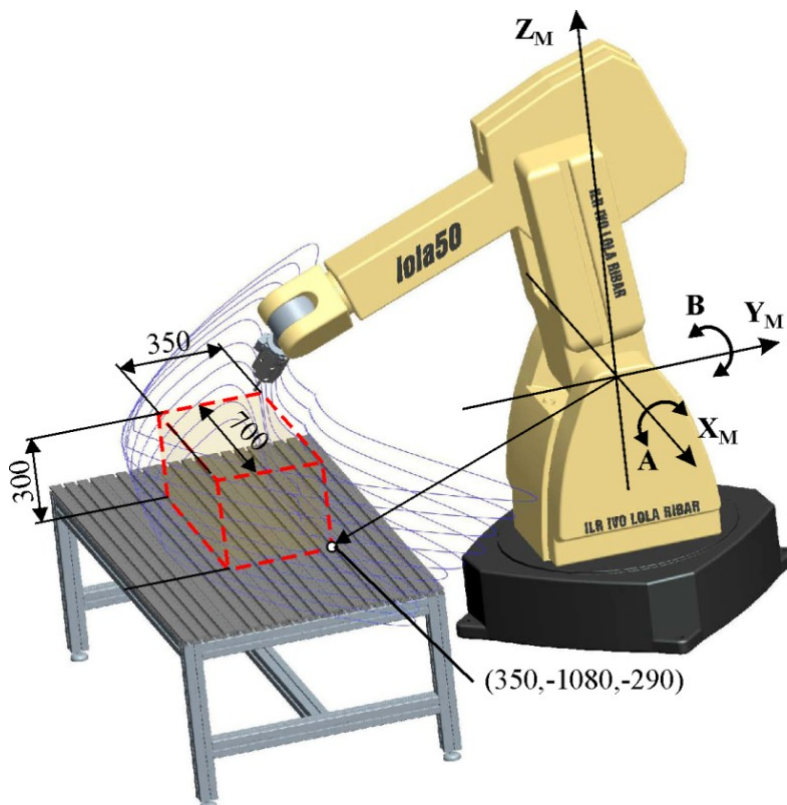
Вредност функције $W_s(X_M, Y_M, Z_M, A_M, B_M)$ је једнака „1“ уколико се позиција и оријентација алата, дефинисана вектором спољашњих координата $x = [x_M \ y_M \ z_M \ A \ B]^T$, може остварити без прекорачења ограничења кретања у зглобовима. Полазећи од изабране тачке у радном простору одређивање радног простора је урађено специфичном стратегијом, корак по корак, која лоцира алат у једном положају и проверава да ли је тај положај могуће остварити или не, узимајући у обзир ограничења кретања зглобова датих на сликама 3.9 и 3.10.

Први корак је одређивање радног простора за случај 3-осне обраде, односно када је оријентација алата фиксна (нпр. $A = 0^\circ, B = 0^\circ$). Део усвојеног радног простора са границама погодним за обраду, $Z_{M_{min}} = -400mm, Z_{M_{max}} = 100mm$, показан је на слици 3.15.

Затим је за усвојен опсег углова оријентације $A[-30^\circ, 30^\circ]$ и $B[-30^\circ, 30^\circ]$ одређен радни простор за случај 5-осне обраде. Део усвојеног радног простора са границама погодним за обраду, $Z_{M_{min}} = -350mm, Z_{M_{max}} = 100mm$, приказан је на слици 3.14.



Слика 3.14 Радни простор елементарне обрадне ћелије на бази робота, која емулира 5-осну глодалицу (X,Y,Z,A,B) , за случај 3-осне обраде



Слика 3.15 Радни простор елементарне обрадне ћелије на бази робота, која емулира 5-осну глодалицу (X,Y,Z,A,B) , за случај 5-осне обраде

Као што је познато из праксе, овај радни простор може бити проширен у случају када су за обраду радног предмета потребне уже границе углова оријентације *A* и *B*. За програмере и операторе CNC машина алатки, оба ова радна простора могу бити редукована на паралелопипеде, као што је то показано на сликама 3.14 и 3.15.

На приказани начин су одређени радни простори свих разматраних конфигурација елементарних обрадних ћелија на бази робота, које емулирају 5-осне хоризонталне и вертикалне глодалице, али због обимности нису овде приказани.

4 РАЗВОЈ МЕТОДА ПРОЈЕКТОВАЊА УПРАВЉАЧКОГ СИСТЕМА ОТВОРЕНЕ АРХИТЕКТУРЕ РЕКОНФИГУРАБИЛНИХ РОБОТСКИХ ЋЕЛИЈА ЗА ОБРАДУ

CNC системи су од појављивања, раних педесетих година, прошли кроз неколико фаза развоја, захваљујући прогресивном напретку хардверских и софтверских технологија. Првобитни *CNC*, заснован на специфичним хардверским решењима, еволуирао је у софтверски који се ослања на универзалну рачунарску платформу и савремене *CNC* софтверске технологије. *PC* рачунарска хардвер који је данас присутан на тржишту, уз савремене *real-time* управљачке софтверске системе и алате отворене архитектуре, даје добру основу за имплементацију софтверски оријентисаног *CNC* система.

Отвореност архитектуре је све присутнија у области управљања роботима и машинама алаткама. На сличан начин на који је утицала на револуционаран раст индустрије персоналних рачунара (*PC*), отвореност може утицати на развој и ширење примене индустријских робота. Иако отворена архитектура (отворен систем) није нови концепт у области софтверског инжењерства, још увек није установљена јединствена дефиниција управљања отворене архитектуре (*OAC-Open Architecture Control*). Многе истраживачке групе, као што су *OMAC (Open Modular Architecture Controllers)* у САД-у, *OSACA (Open System Architecture for Controls within Automation System)* у Е.У. и *JOP (Japanese Open Promotion Group)* у Јапану, имају свој став према поменутиим системима. Према *IEEE 1003.0 (Institute of Electrical and Electronics Engineers)* моделу, отворени систем је дефинисан као систем који омогућава да се апликације имплементирани на одговарајући начин могу извршавати на платформама различитих произвођача, комуницирати са другим системским апликацијама и омогућити конзистентни начин интеракције са корисником.

Са становишта управљања, софтверски системи отворене архитектуре постају значајан део производних система у којима корисници и интегратори система директно могу утицати на управљачку конфигурацију у тренутку

када то захтева сам производни процес.

Управљачки системи отворене архитектуре се могу сврстати у три категорије:

- Са отвореним корисничким интерфејсом;
- Са ограниченом отвореношћу управљачког језгра;
- Са потпуно отвореном архитектуром.

У табели 4.1 су приказане различите структуре и приступи развоју хардвера и софтвера управљачких система машина алатки и робота.

Табела 4.1 Структуре управљачких система машина алатки и робота

	(i) Стандардан интерфејс ка аналогним актуаторима	(ii) Управљање концентрисано на актуаторском нивоу, дигитални интерфејс ка актуаторима	(iii) Управљање концентрисано на корисничком нивоу (софтверски CNC)
Процесор 1: ниво корисника	HMI, NC i PLC програмски систем	HMI, NC i PLC програмски систем	HMI, NC i PLC програмски систем, планер путање, контролер процеса, контролер позиције
Процесор 2: ниво управљања	планер путање, контролер позиције		
Процесор 3: ниво управљања		PLC	
Процесор 4: ниво управљања	контролер процеса		
Процесор 5: ниво актуатора	контролер брзине, струјни контролер, појачавач снаге	планер путање, контролер процеса, контролер позиције + контролер брзине, струјни контролер, појачавач снаге	контролер брзине, струјни контролер, појачавач снаге

У варијанти (i) уз аналогни интерфејс на страни актуатора, сваки извршни модул поседује свој процесор, што води ка коришћењу специфичног хардвера за различите процесе.

У варијанти (ii) приказана је структура управљања која интегрише интелигентне дигиталне погоне високих перформанси, што доприноси

мањој комплексности система.

У варијанти (iii) приказан је управљачки систем заснован на *PC* управљачкој архитектури и оперативном систему за рад у реалном времену. У овој варијанти све контролне функције се извршавају као софтверски процеси на *PC real-time* платформи.

Комбиновање софтверских модула води ка значајном смањењу броја процесора. То, поред већ описаних предности *OAC*-а, повлачи за собом ниску цену хардвера управљачке јединице и намеће правац у развоју управљачког система отворене архитектуре за роботе и машине алатке.

Као што је већ поменуто, идеја да се постојећи *CAD/CAM* системи користе за програмирање робота у *G*-кôду за вишеосну обраду резањем, наводи на приступ развоју управљачког система који треба кориснику да омогући емуляцију хоризонталних и вертикалних 5-осних машина алатки, роботима вертикалне зглобне конфигурације. С обзиром на чињеницу да управљачки систем својом реконфигурабилношћу треба да покрије све већ описане конфигурације роботских ћелија за обраду, слике 3.1 до 3.7, али и да омогући интеграторима система да сами креирају нове конфигурације, одговарајући софтверски управљачки модули, као што су кинематички, модули интерфејса и сл., морају бити доступни за модификацију и/или проширење. Уз чињеницу да могућност конфигурисања и реконфигурисања управљања зависи од отворености, модуларности и скалабилности управљачке архитектуре [29], у овој дисертацији су постављене су две методе пројектовања реконфигурабилних управљачких система које се заснивају на софтверски оријентисаном приступу, чиме се омогућава реконфигурисање система и од стране корисника.

Метод на бази примене готових хардверских и софтверских система отворене архитектуре

Метод пројектовања управљачког система роботских ћелија за обраду, применом готових хардверских и софтверских решења отворене архитектуре, заснива се на:

- Једном од постојећих управљачких софтверских система отворене

архитектуре, *EMC2 (Enhanced Machine Controller* верзија 2), *KCAM, TurboCNC, Mach3, DeskCNC, CNCZeus, DesKAM, CNCPro* или сл., који се интегрише у управљачки систем без измена изворног кода;

- Интерпретеру *G*-кода, интегрисаном у управљачки софтвер;
- Могућности имплементације сопствених кинематичких модула, која проистиче из отворености архитектуре управљачког софтвера;
- Могућности конфигурисања виртуелних емулираних машина алатки, односно работа посредством одговарајућих графичких библиотека;
- Постојању конфигурационог интерфејса који омогућава реконфигурирање система;
- Оперативном систему отворене архитектуре за рад у реалном времену;
- Рачунарској хардверској платформи отворене архитектуре.

Основна предност примене готових софтверских решења у реализацији управљачког система за обрадне ћелије на бази робота се огледа у могућности прецизне предикције перформанси и времена реализације управљања у раним фазама развоја. Са друге стране, недостатак оваквог приступа се огледа у ограниченим могућностима реконфигурисања реализованог система од стране корисника, као и ограниченим могућностима примене одговарајућих сензора у циљу повећања тачности обраде и побољшања безбедности оператора током рада на роботској ћелији за обраду.

Рад на овој дисертацији је обухватио развој и комплетну имплементацију реконфигурабилног управљачког система, применом расположивог софтвера отворене архитектуре *EMC2* [78], *Ubuntu Linux* оперативног система са *real-time* језгром и *PC* хардвера, као готових решења.

Метод на бази примене софтверских алата и хардверских технологија отворене архитектуре уз развој алгоритама управљања.

Метод пројектовања управљачког система роботских ћелија за обраду, који се заснива на сопственом развоју управљачких алгоритама, представља флексибилније решење за интеграторе система и кориснике. Применом овог

метода се обезбеђује статичка и динамичка реконфигурабилност система, али и захтева обиман програмерски рад у циљу развоја управљачких алгоритама, конфигурационих софтверских алата и система за верификацију управљачких конфигурација. Према овом приступу, општи концепт развоја управљачког система се заснива на:

- Сопственом развоју компилатора програма обраде (*G*-кôда, *STEP-NC*-а);
- Сопственом развоју интерполатора;
- Имплементацији развијених кинематичких модула;
- Примени софтверских алата отворене архитектуре за креирање и синхронизацију управљачких модула или реализацију сопствених;
- Примену софтверских алата отворене архитектуре за одабир и повезивање управљачких модула, односно конфигурисање и реконфигурисање система или реализацију сопствених;
- Оперативном систему рачунара отворене архитектуре за рад у реалном времену;
- Рачунарској хардверској платформи отворене архитектуре.

Развој управљачког система на бази софтверских алата и хардверских технологија отворене архитектуре, уз развој алгоритама управљања, омогућава прецизно испуњење захтева корисника, како у погледу перформанси система, тако и по питању изгледа и структуре корисничких интерфејса, као и верификационих софтверских алата.

Како процедура реконфигурисања мора бити праћена одговарајућим тестовима и симулацијама у циљу потврде функционалности и безбедности новоформиране конфигурације, уз конфигурациони интерфејс и одговарајући верификациони софтверски алати морају бити интегрални део реконфигурабилног управљачког система.

Рад на овој дисертацији је обухватио развој нових и прилагођење претходно развијених управљачких алгоритама, имплементацију дела софтверских модула применом софтверских алата које нуди *OROCOS (Open Robot COntrol Software)* [79], односно *Linux*, као и реализацију првог прототипа елементарне конфигурације управљачке јединице на *PC* хардверу у циљу тестирања основних модула за управљање кретањима.

4.1 Развој и имплементација реконфигурабилног управљачког система на бази готовог хардвера и софтвера отворене архитектуре применом софтвера *EMC2*

Као што је већ поменуто, истраживања у области развоја реконфигурабилних обрадних система на бази робота, која се спроводе на Катедри за производно машинство машинског факултета у Београду и Лола институту, у којима учествује и аутор, одвијају се у два правца (*i*) развој реконфигурабилног система управљања и програмирања обрадних ћелија на бази робота у *G*-коду, (*ii*) развој алгоритама за повећање тачности обраде. Прва фаза истраживања је обухватила:

- Развој реконфигурабилног управљачког система на бази готових хардверских и софтверских решења отворене архитектуре на примеру примене *EMC2* софтвера, који је део ове дисертације;
- Развој сопствених алгоритама и система за *off-line* компензацију грешака у обради насталих као последица статичке попустљивости робота [27, 28].

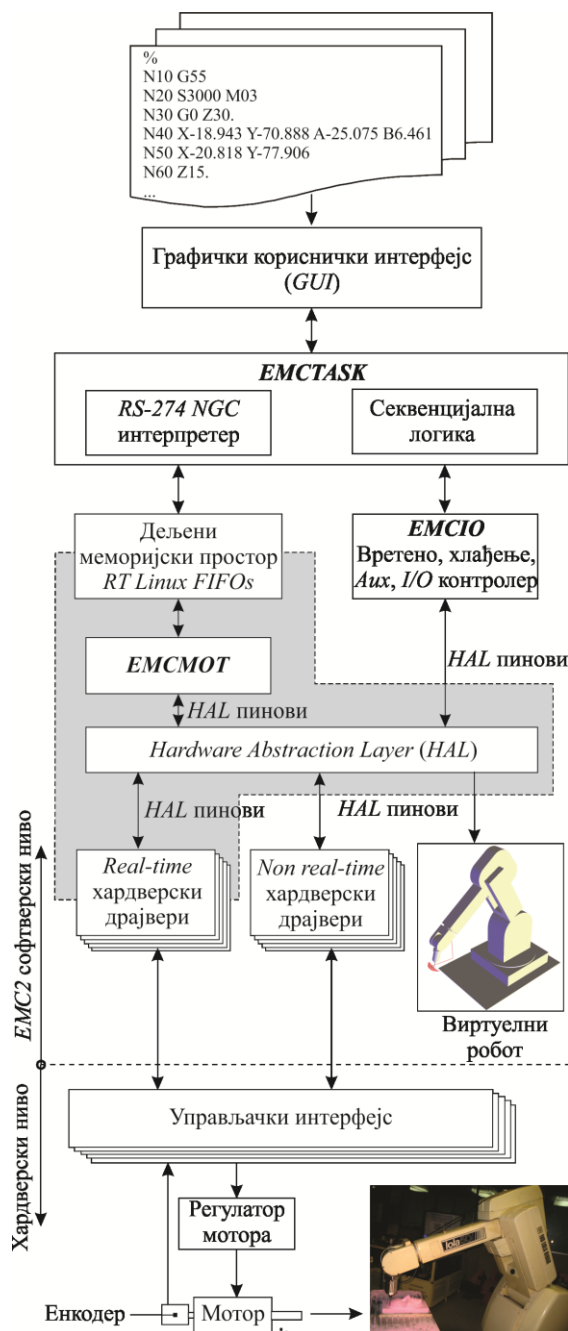
Резултати ових истраживања су имплементирани у два софтверска система, која функционишу независно на различитим рачунарским платформама са идејом да се у каснијим фазама развоја интегришу у јединствен систем.

Део активности Националног института за стандарде и технологију САД-а, познатијег као *NIST (National Institute of Standards and Technology)*, усмерен је на развој мерних метода и стандарда за интелигентне управљачке системе у производњи. Као резултат дугогодишњег истраживања у овој области, *NIST* је промовисао *RCS (Real-Time Control System)* софтверску библиотеку [80], која данас представља стандардну, референтну основу за развој интелигентних управљачких система. *RCS* библиотека је еволуирала током година и резултирала развојем бројних управљачких апликација, укључујући и *EMC2*, *real-time* управљачки софтвер отворене архитектуре, који се може искористити за управљање најразличитијих машина алатки и робота. Обиман рад и искуство који су годинама улагани у *RCS* и *EMC*, у светским

размерама представљају добру технолошку основу за имплементацију сопственог система управљања и програмирања машина алатки и робота.

4.1.1 Структура софтвера EMC2

Упростиена структура реконфигурабилног система за управљање и програмирање реконфигурабилних роботских ћелија за обраду на бази EMC2 софтвера приказана је на слици 4.1.



Слика 4.1 Архитектура EMC2

Као што се са слике 4.1 може видети, *EMC2* софтвер садржи четири основна модула:

- Контролер кретања *EMCMOT*;
- Контролер улазно/излазних дискретних сигнала *EMCIO*;
- Контролер процеса *EMCTASK*;
- Графички кориснички интерфејс *GUI (Graphical User Interface)*.

EMCMOT је софтверски модул који ради у реалном времену. Задужен је за интерполацију програмиране путање, прорачун директне и инверзне кинематике, као и генерисање одговарајућих вредности сигнала који се предају регулаторима мотора у sukcesивним временским тренуцима. Кинематички модули, који улазе у састав контролера кретања, базирани су на генерализованом кинематичком моделу разматраних конфигурација обрадних ћелија на бази робота датих у поглављу 3. Добијена решења директне и инверзне кинематике су виду одговарајућих функција програмирана у *C* језику и интегрисана у контролер кретања.

EMCIO модул процесира све операције над периферијама које нису у директној вези са моторима за погон оса машине.

EMCTASK модул је одговоран за интерпретирање инструкција *G*-кода и координацију активности модула *EMCMOT* и *EMCIO*. Ове активности су садржане у програму обраде, односно низу инструкција *G* и *M* кода, које контролер процеса интерпретира и у одговарајућим временским тренуцима прослеђује контролеру кретања или улазно/излазном контролеру дискретних сигнала. Као што се са слике 4.1 може видети, комуникација између *real-time* *EMCMOT* модула и *non-real-time* модула *EMCTASK* се обавља преко дељеног меморијског бафера посредством *RTLinux FIFO* механизма.

GUI спада у екстерне програмске модуле и представља графички кориснички интерфејс за преузимање и покретање програма обраде, иницијализацију као и праћење рада машине посредством графичког приказа путање алата и виртуелног роботског окружења. Од неколико графичких корисничких интерфејса, који се налазе у саставу *EMC*-а, приликом конфигурисања управљачког система изабран је *AXIS* интерфејс,

који је конфигуриран према специфичним потребама управљања реконфигурабилним обрадним ћелијама на бази робота.

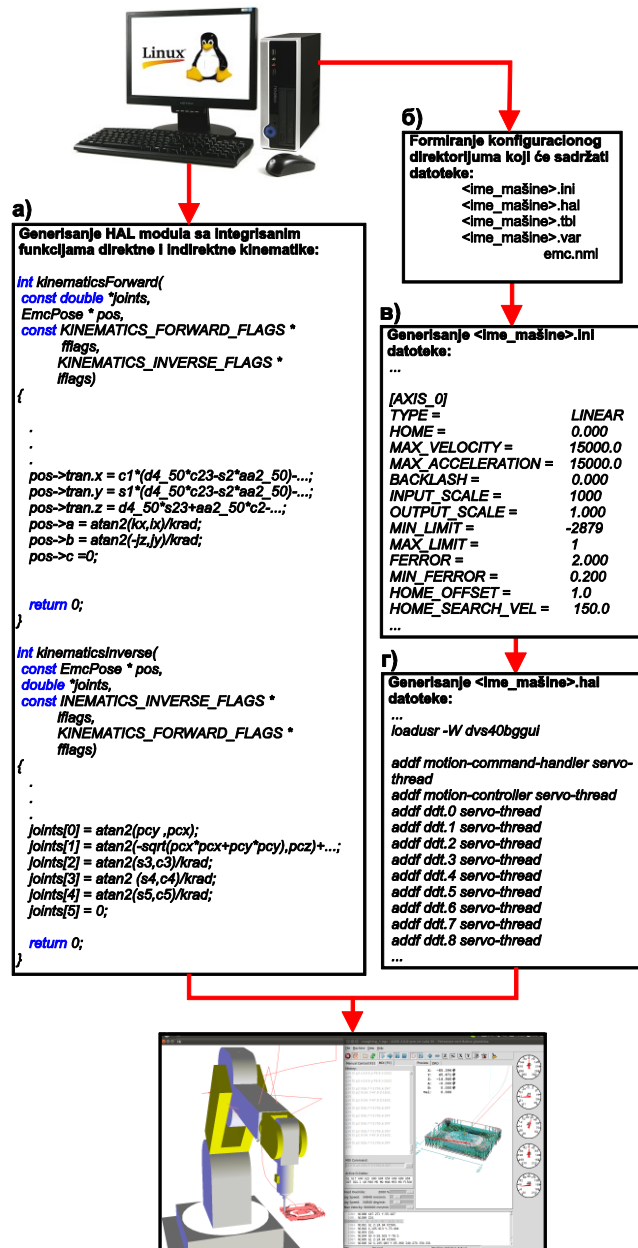
EMC је током година постојања, од софтвера намењеног ентузијастима, еволуирао у управљачки софтвер присутан, како у академским, тако и у индустријским применама. Томе је значајно допринео развој *HAL*-а (*Hardware Abstraction Layer*) који омогућава једноставну интеграцију функција инверзне и директне кинематике у систем за управљање, као и униформни интерфејс за повезивање са хардверским и софтверским модулима система. Значајан допринос популаризацији *EMC*-а је имало проширење графичких интерфејса приказом комплетне машине у виртуелном *3D* окружењу, што је омогућило формирање система за *off-line* симулације, посебно значајне за примене код робота.

Развој експерименталног прототипа управљачког система на бази *EMC2* за управљање обрадним ћелијама на бази робота са 5 и 6 степени слободе који емулирају 5-осне машине алатке је базиран на методологији конфигурисања *EMC2* софтвера приказаној на слици 4.2 и обухвата:

- Креирање датотека на бази функција инверзне кинематике, нпр. за конфигурацију са слике 4.2а, једначине (3.107) до (3.125), и директне кинематике, једначине (3.60) до (3.89), обрадних ћелија на бази робота вертикалне зглобне конфигурације и генерисање одговарајућих *HAL* модула;
- Формирање конфигурационих директоријума за сваку конфигурацију обрадне ћелије на бази робота вертикалне зглобне конфигурације, слика 4.2б;
- Дефинисање конфигурационих **.ini* датотека које садрже параметре робота и додатних ротационих и транслаторних оса, као су: опсеги кретања радних оса, максималне и радне брзине и убрзања за поједине осе, параметри позиционих *PID* (*Proportional-Integral-Derivative*) регулатора, параметри за рад система у реалном времену, секвенце иницијализације итд, слика 4.2б;
- Дефинисање конфигурационих **.hal* датотека које садрже елементе за генерисање позиционих серво петљи, параметре *HAL* интерфејса ка

роботима, реалним или виртуелним, елементе неопходне за иницијализацију робота итд, слика 4.2г;

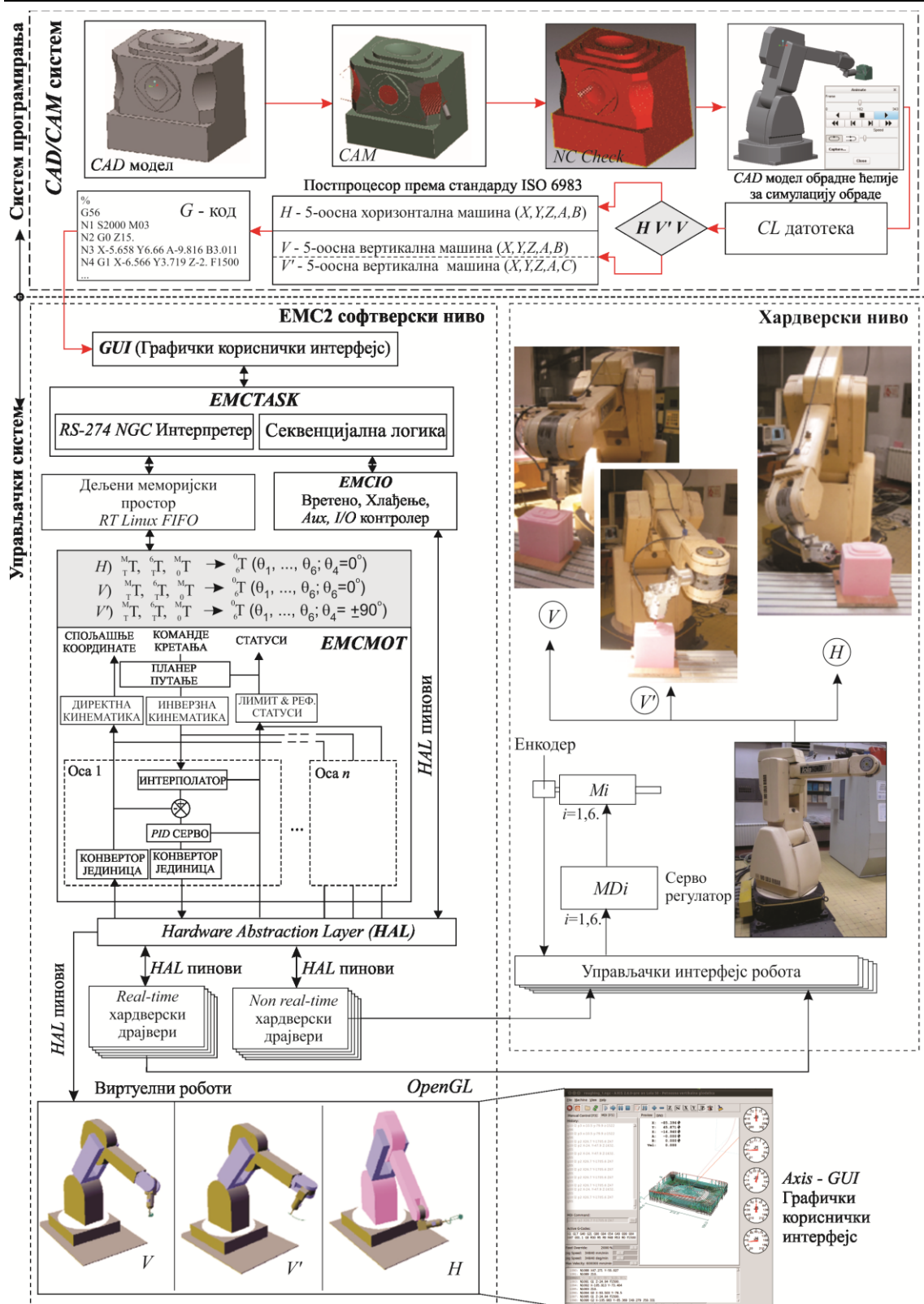
- Конфигурисање хардвера управљачког на *PC* рачунарској платформи која мора бити проширена одговарајућим интерфејс картицама за улазно/излазне операције.



Слика 4.2 Метод конфигурисања *EMC2*

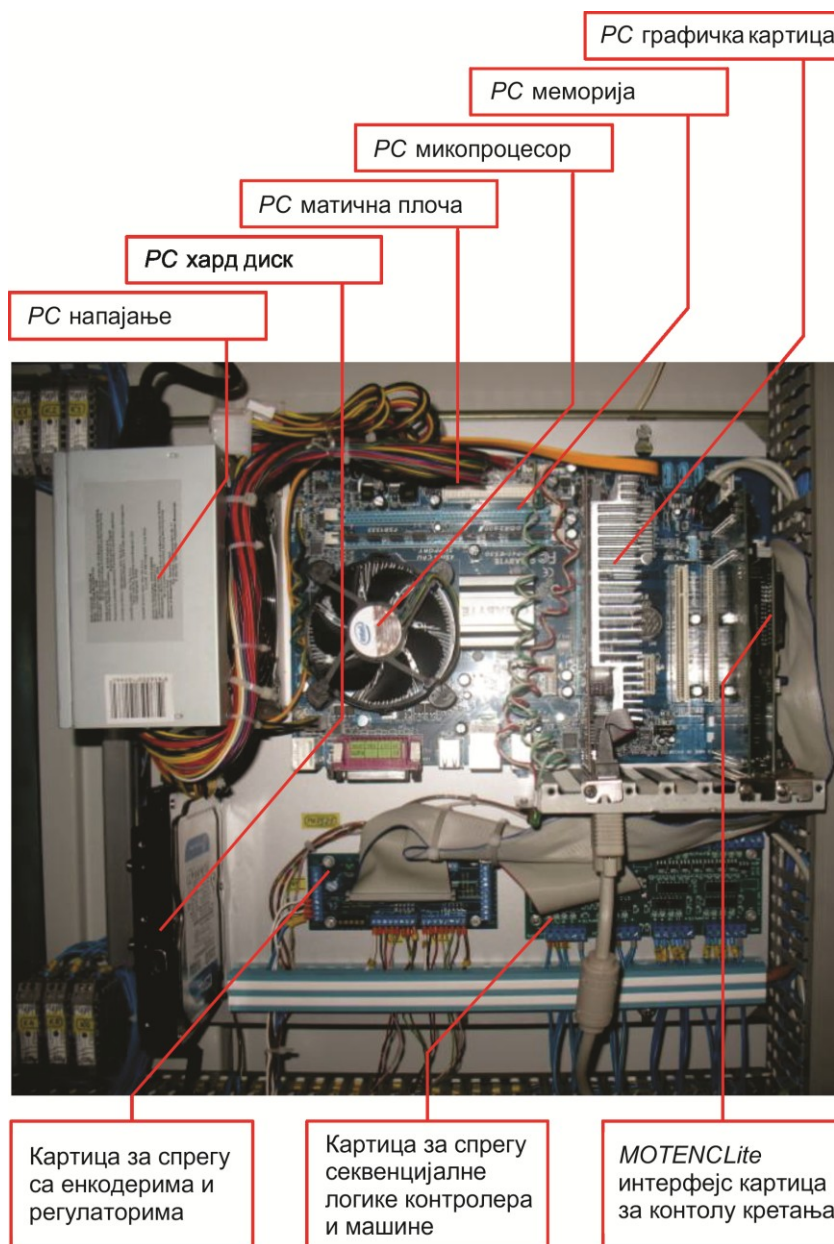
Структура експерименталног прототипа реконфигурабилног система управљања и програмирања елементарне обрадне ћелије на бази робота вертикалне зглобне конфигурације са 6 степени слободе, реализованог применом *EMC2* софтвера, приказана је на слици 4.3.

РАЗВОЈ МЕТОДА ПРОЈЕКТОВАЊА УПРАВЉАЧКОГ СИСТЕМА...



Слика 4.3 Упрощена структура реконфигурабилног система за управљање и програмирање реконфигурабилних роботских ћелија за обраду на бази EMC2 софтвера

Пример реализованог експерименталног прототипа управљачке јединице, на бази *EMC2* софтвера конфигурисаног на претходно описани начин показан је на слици 4.4.



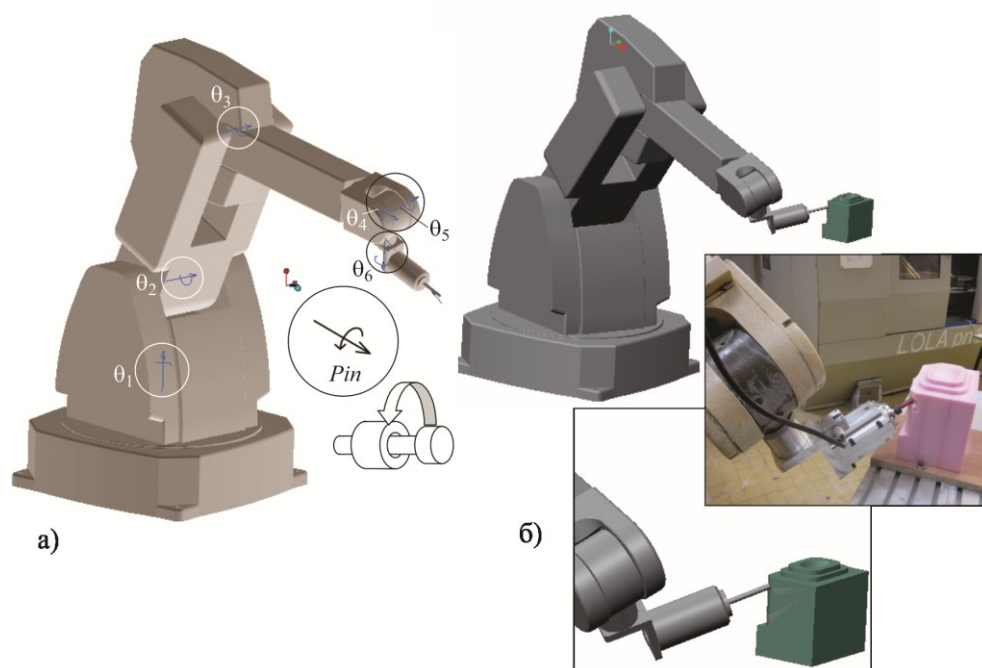
Слика 4.4 Хардвер развијеног експерименталног прототипа управљачке јединице отворене архитектуре

4.1.2 Систем за програмирање и симулацију реконфигурабилних обрадних ћелија на бази робота у *CAD/CAM* окружењу

Програмирање реконфигурабилних обрадних ћелија на бази робота се обавља на конвенционалан начин у стандардним *CAD/CAM* окружењима. Као

што је показано на слици 4.3, полази се од радног предмета који се моделира посредством одговарајућег *CAD* система и одговарајућег припремка. У *CAM* систему се прво врши учитавање референтног *CAD* модела радног предмета са изабраним координатним системом за програмирање $\{W\}$, као и дефинисање димензија одговарајућег припремка. Следећа активност се односи на примену различитих расположивих *CAM* стратегија и генерисање путање алата (*CL* датотека). Како графичка симулација представља изузетно важан елемент у програмирању робота уопште, то и у случају примене робота за обраду чини интегрални део система управљања и програмирања. Верификацију програма обраде у *CAD/CAM* систему програмер остварује на следеће начине: симулацијом путање алата, симулацијом уклањања материјала (*NC check*) и на *CAD* моделу обрадне ћелије на бази робота која ради по задатом програму обраде, користећи стандардне кинематичке модуле *CAD*-а за нумеричко решавање инверзног кинематичког проблема.

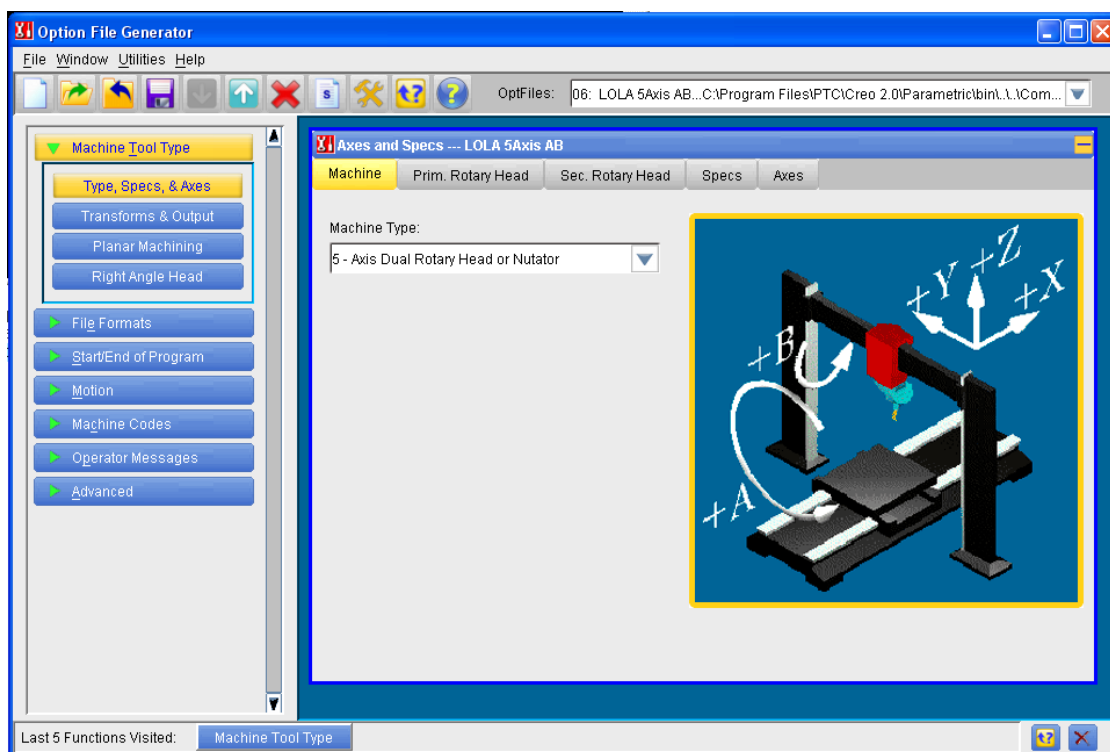
Обрадна ћелија за симулацију обраде у *CAD*-у, слика 4.5а, је моделирана у размери 1:1 и садржи *CAD* модел робота и *CAD* моделе додатних обртних и/или транслаторних оса, укључујући и кинематичке везе између њихових покретних сегмената.



Слика 4.5 Упоредни приказ виртуелне обрадне ћелије на бази робота и реалног робота током обраде

Коришћењем стандардног *CAD* модула за симулацију кретања механизма, који укључује и функције за нумеричко решавање инверзног и директног кинематичког проблема, током симулације се узимају у обзир ограничења кретања у зглобовима оса робота, помоћних транслаторних и/или обртних оса и омогућава визуелна детекција могућих колизија, слика 4.5б.

Након верификације путање алата (*CL* датотеке) следи поступак постпроцесирања и добијање *G*-кода. Пре поступка постпроцесирања потребно је конфигурисати одговарајуће постпроцесоре за 5-осне машине алатке конфигурација (X, Y, Z, A, B) и (X, Y, Z, A, C) . Конфигурисање се врши помоћу конфигуратора постпроцесора у *CAD/CAM* окружењу, слика 4.6. Овако добијени програм обраде у *G*-коду се преноси у управљачку јединицу обрадне ћелије на бази робота.



Слика 4.6 Пример конфигурисања постпроцесора за 5-осну машину алатку конфигурације (X, Y, Z, A, B) чији рад емулира обрадна ћелија на бази робота

4.1.3 Конфигурисање виртуелних реконфигурабилних ћелија за обраду на бази робота

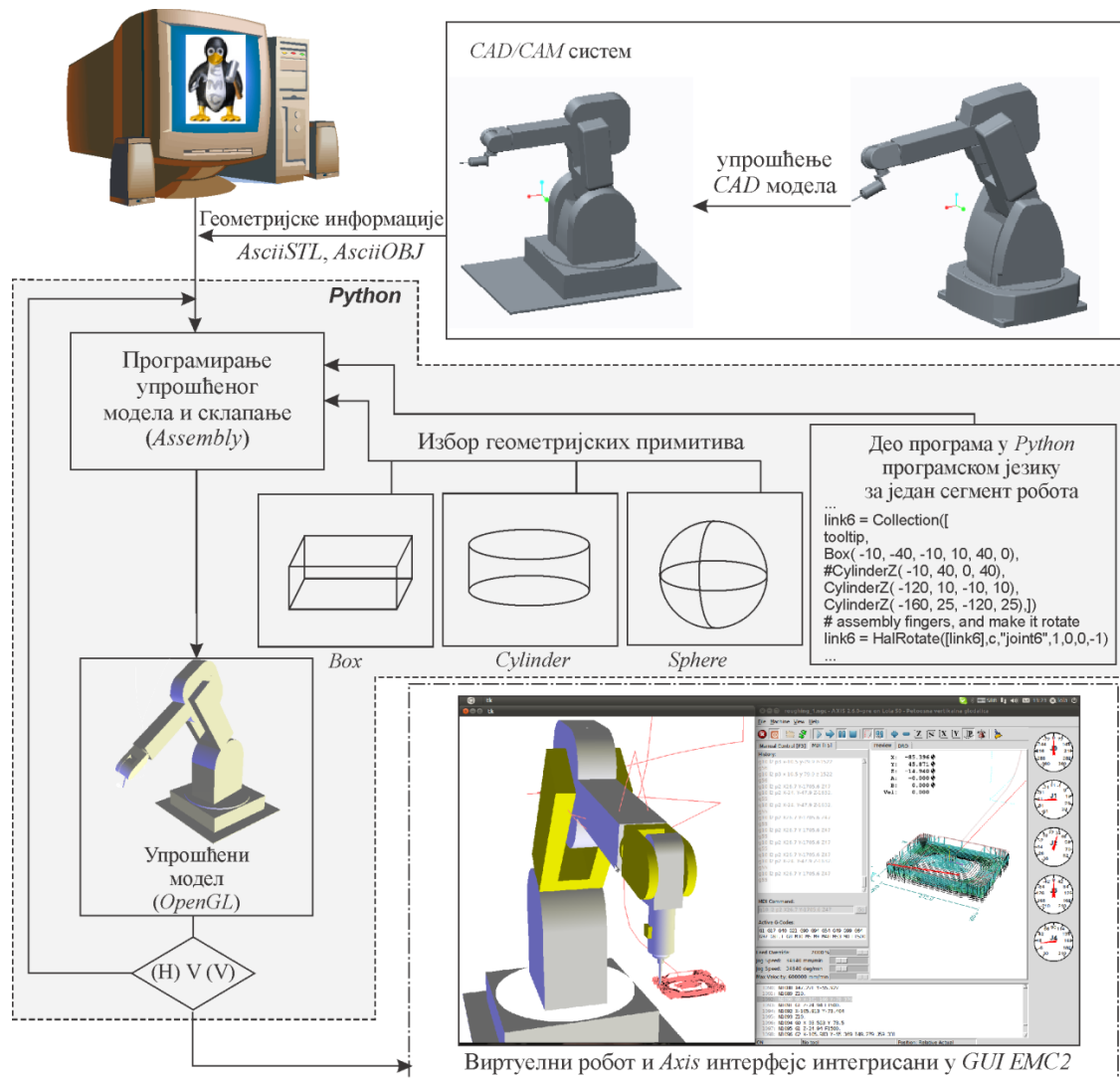
Како графичка симулација представља изузетно важан елемент у програмирању робота уопште, као што је то већ наглашено, поред верификације програма обраде симулацијом у *CAD/CAM* окружењу, неопходно је извршити верификацију одабране конфигурације обрадне ћелије коришћењем њеног виртуелног модела. Узимајући у обзир овакав захтев, развијен је систем управљања и програмирања који омогућава, како *off-line* симулацију програма обраде у *G*-кôду, тако и праћење рада робота у реалном времену посредством конфигурисаних *3D* виртуелних модела робота, односно ћелија за обраду. Овакав приступ обезбеђује да се, пре обраде на реалној роботској ћелији, могу урадити следеће провере:

- Да ли је радни предмет коректно постављен унутар радног простора;
- Да ли робот може да изврши програмирану трајекторију без прекорачења опсега кретања унутрашњих координата;
- Визуелна детекција колизија између сегмената робота као и алата и помоћног прибора.

Виртуелни модели робота, односно обрадних ћелија, су развијени у програмском језику *Python* уз коришћење *OpenGL*-а (*Open Graphics Library*), слика 4.7. Поступак моделирања се своди на програмирање координата, за дефинисање елементарних геометријских примитива. Да би се посао олакшао, најпре се дефинише упрошћени модел робота у неком *CAD* пакету, одакле се могу преузети потребне координате геометријских примитива виртуелног робота, односно додатних оса. Нова могућност *EMC2*, која у многоме олакшава посао, је интеграција виртуелних сегмената машине који су претходно моделирани у неком стандардном *CAD*-у, а затим извезени у облику **.stl* или **.obj* датотеке.

За сваку конфигурацију обрадне ћелије на бази робота, предложену у поглављу 2 ове дисертације, моделирана је виртуелна обрадна ћелија, слика 4.8. Приликом моделирања су коришћени прости геометријски примитиви, а зглобови виртуелних робота су управљани сигналима који се користе и за управљање кретањима оса реалног робота.

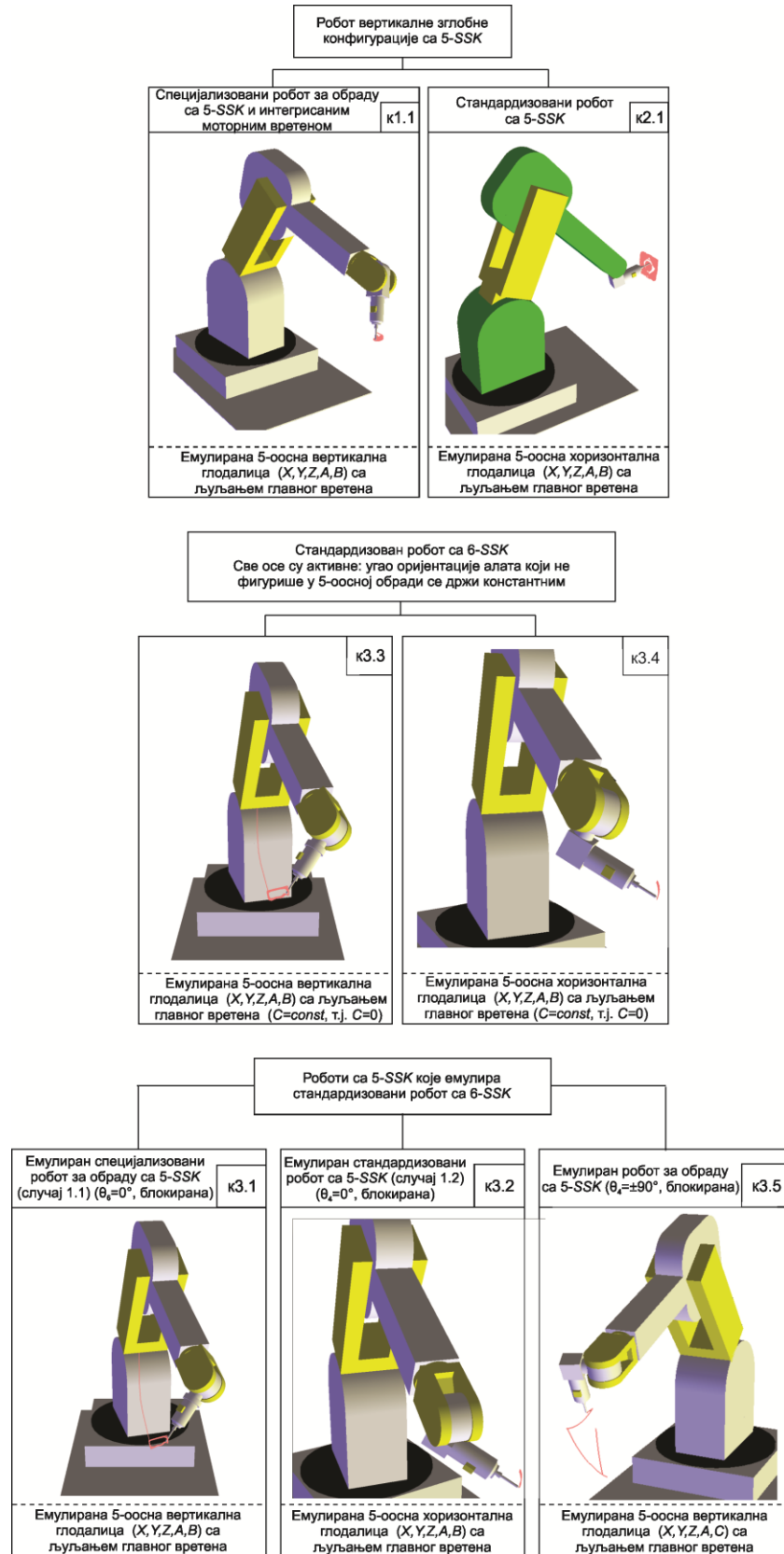
РАЗВОЈ МЕТОДА ПРОЈЕКТОВАЊА УПРАВЉАЧКОГ СИСТЕМА...



Слика 4.7 Конфигурисање виртуелних модела у EMC2

Симулације рада реконфигурабилних обрадних ћелија на бази робота имају за циљ тестирање сваке конфигурације обрадног система на бази робота (конфигурације к1.1, к1.2, к2.1, к2.2, к3.1, к3.2 и к3.3), што обухвата проверу имплементираних кинематичких алгоритама (функција инверзне и директне кинематике програмираних на програмском језику C/C++), функционалности софтверских интерфејса ка хардверу обрадног система и новоформираних GUI-а.

По извршеним симулацијама на одабраној конфигурацији виртуелног обрадног система на бази робота, верификован програм обраде се може безбедно извршавати на реалној обрадној ћелији на бази робота, односно емулираној 5-осној машини алатки.



Слика 4.8 Виртуелни 5-осни обрадни системи на бази робота

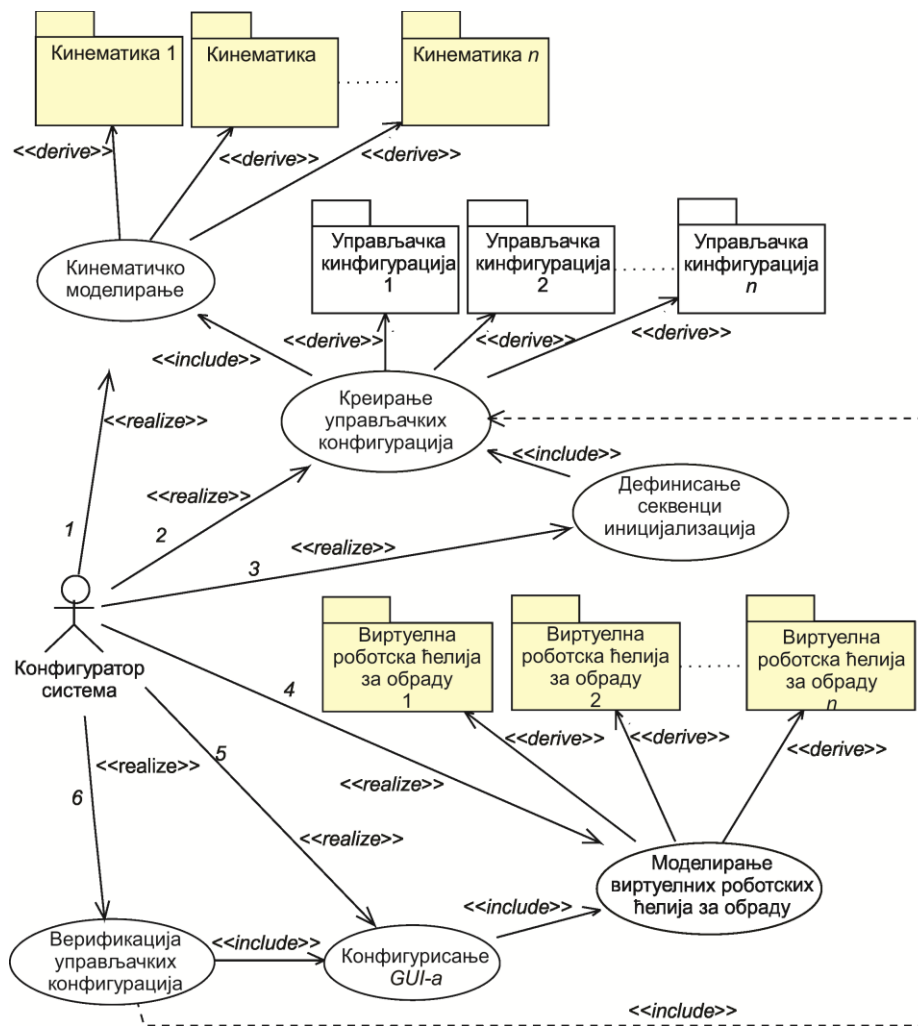
Виртуелни 3D модел емулиране машине повезан са *AXIS* интерфејсом и *EMC2* софтвером омогућава *off-line* програмирање са тестирањем и верификацијом програма обраде на удаљеном програмерском месту, без ангажовања самог робота. Рад у *off-line* виртуелном окружењу је погодан и са аспекта обуке и едукације програмирања.

Приликом тестирања сваке управљачке конфигурације, као и при тестирању програма обраде, управљачки сигнали се посредством *HAL*-а у реалном времену прослеђују конфигурираној виртуелној роботској ћелији за обраду. Касније у току обраде, идентични управљачки сигнали се посредством *HAL*-а и хардверског интерфејса за улазно/излазне операције, који укључује *ADC (Analog-to-Digital Converter)*, *DAC (Digital-to-Analog Converter)* и улазно/излазне канале, усмеравају ка регулаторима серво мотора.

4.1.4 Приказ морфолошког метода за конфигурисање и реконфигурисање управљачког система на бази *EMC2*

Да би управљачки систем на бази *EMC2* могао бити конфигуриран за управљање различитим конфигурацијама обрадних ћелија на бази робота вертикалне зглобне конфигурације, мора се формирати коначан број кинематичких модела, односно софтверских модула. Реконфигурабилни управљачки систем мора укључити све софтверске компоненте и кинематичке конфигурације потребне за управљање одабраним конфигурацијама обрадне ћелије на бази робота. Процедура за конфигурисање реконфигурабилног система управљања развијеног на бази *EMC2* софтвера обавља се према успостављеној методологији приказаној применом *UML (Unified Modeling Language [81])* дијаграма на слици 4.9. Према постављеној методи за конфигурисање управљачког система реконфигурабилних ћелија на бази робота у овој дисертацији, систем интегратор, односно произвођач, иницијално поставља скуп изабраних конфигурација (нпр. к3.1...к3.5) за које треба да изврши следеће активности:

- Решавање директне и инверзне кинематике за одабране конфигурације робота за обраду, на начин који је приказан у поглављу 3;



Слика 4.9 UML дијаграм конфигурисања управљачког система на бази EMC2

- Креирање одговарајућег броја кинематичких софтверских модула за одабране конфигурације робота за обраду на основу претходно добијених решења инверзног и директног кинематичког проблема;
- Имплементација конфигурационих датотека са релевантним параметрима за сваку кинематичку конфигурацију;
- Дефинисање иницијализационе секвенце за сваку конфигурацију обрадне ћелије на бази робота;
- Конфигурисање виртуелних роботских ћелија за одабране конфигурације обрадне ћелије на бази робота;
- Конфигурисање графичког корисничког интерфејса (GUI) у складу са специфичностима обраде роботом и захтевима корисника (дефинисање графичких елемената за праћење позиција оса робота, покретање и праћење процеса иницијализације, контролу брзине

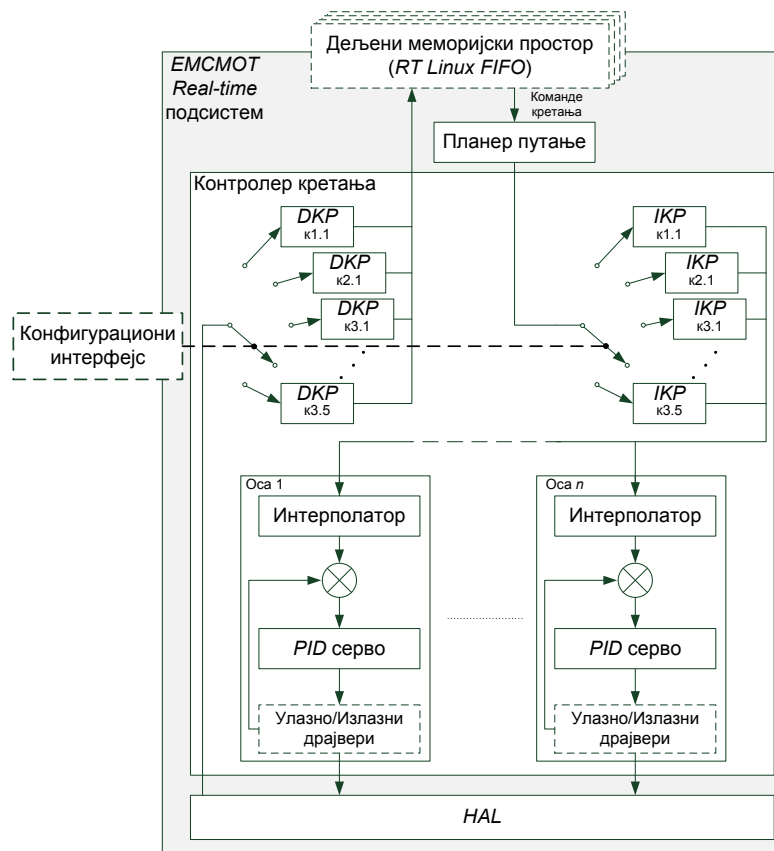
помоћног кретања итд.);

- Тестирање одабраних конфигурација обрадне ћелије на бази робота посебно израђеним тест програмима обраде, најпре у претходно конфигурисаним виртуелним окружењима, а после верификације и на реалним конфигурацијама.

Реконфигурабилност управљачког система на бази *EMC2* се заснива на успостављању одговарајућих веза између дефинисаних софтверских модула посредством апликативних интерфејса које обезбеђује *HAL*. Том приликом, према инструкцијама садржаним у конфигурационим датотекама, врши се повезивање одговарајућих кинематичких модула са модулима за планирање и контролу кретања система за рад у реалном времену, слика 4.10.

Структура *GUI*-а са типом и распоредом софтверских модула које укључује, слика 4.3, је такође дефинисана у једној од конфигурационих датотека.

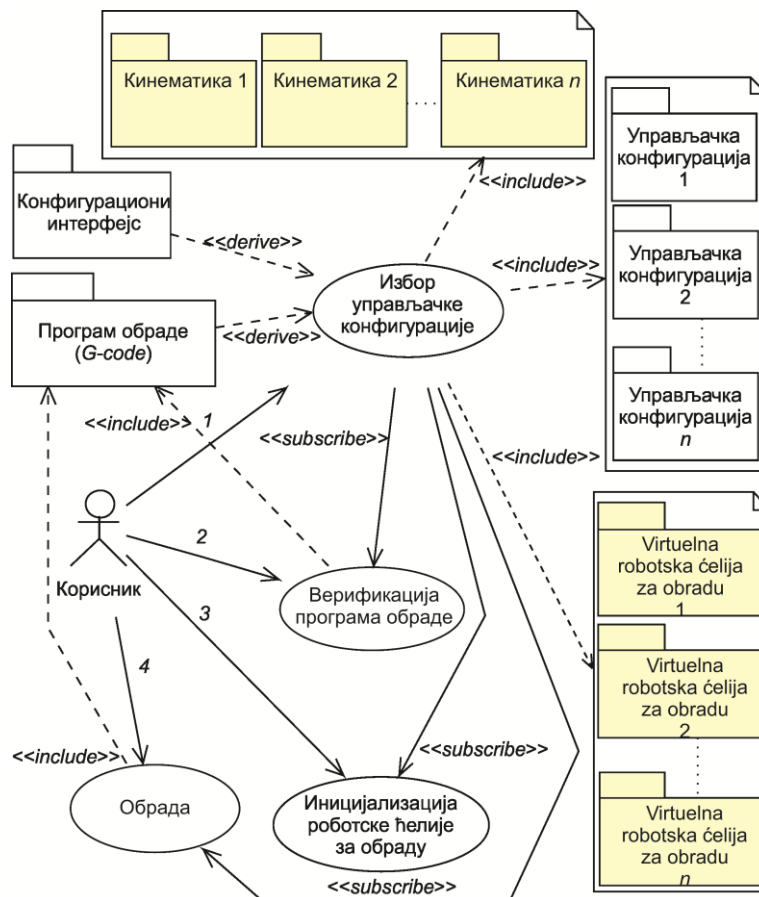
Процедура за реконфигурисање система управљања развијеног на бази *EMC2* софтвера обавља се према успостављеној методологији приказаној



Слика 4.10 Символички приказ начина реконфигурисања

применом *UML* дијаграма на слици 4.11. Под претпоставком да су кинематичке конфигурације дефинисане и да је систем иницијално конфигуриран, реконфигурирање управљачког система подразумева обављање следећих активности према успостављеној методологији:

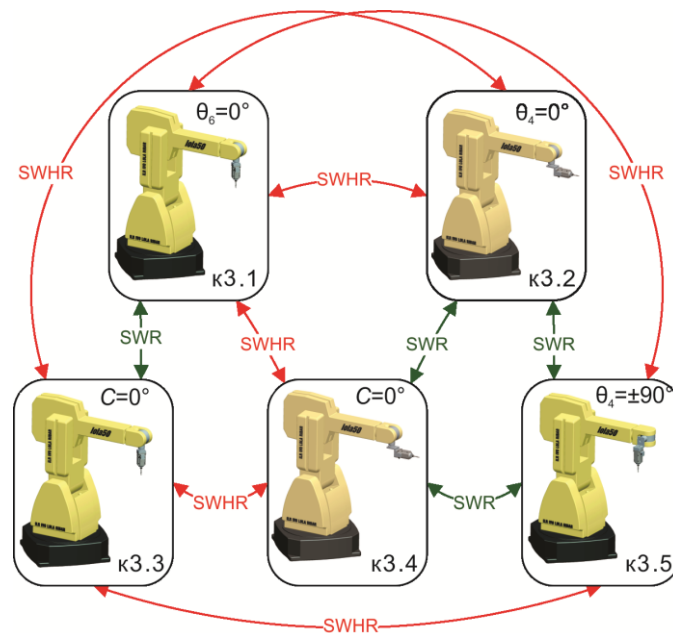
- Избор одговарајућег кинематичког модула, конфигурационе датотеке и конфигурације виртуелне обрадне ћелије на бази робота, посредством конфигурационог интерфејса управљачког система, за потребе верификације конфигурисаног система и програма обраде;
- Верификацију програма обраде на одговарајућој конфигурацији виртуелне обрадне ћелије уз симулацију путање алата и праћење сегмената робота ради уочавања евентуалних колизија;
- Иницијализација обрадне ћелије на бази робота у складу са одабраном кинематичком конфигурацијом;
- Обрада радног предмета претходно верификованим програмом обраде, *G*-кôдом.



Слика 4.11 *UML* дијаграм реконфигурирања управљачког система на бази *EMC2*

Као илустративни пример реконфигурисања управљачког система, на слици 4.12 је приказан дијаграм реконфигурисања за случај реконфигурабилне обрадне ћелије на бази стандардизованог робота са 6 степени слободe у конфигурацијама к3.1, к3.2, к3.3, к3.4 и к3.5., слике 3.3 до 3.7. Реконфигурација система, зависно од почетне и захтеване конфигурације, може бити остварена на два начина:

- Софтверски (*SWR*), када није потребна интервенција на хардверу система, нпр приликом преласка са конфигурације к3.4, слика 3.6, на конфигурацију к3.5, слика 3.7, или са конфигурације к3.1, слика 3.3, на конфигурацију к3.3, слика 3.5. Овај начин реконфигурисања је посебно значајан када је потребно остварити вишестрану обраду сложених радних предмета у само једном базирању, при чему обрадна ћелија на бази робота може емулирати рад и вертикалне и хоризонталне 5-осне машине алатке као што је показано на слици 2.20;
- Софтверско-хардверски (*SWHR*), када је потребно учинити прилагођење (реконфигурисање) софтвера и хардвера обрадне ћелије на бази робота, односно промену механичког интерфејса између моторног вретена и стандардног интерфејса робота, што је неопходно учинити како би била омогућена емулација одговарајуће

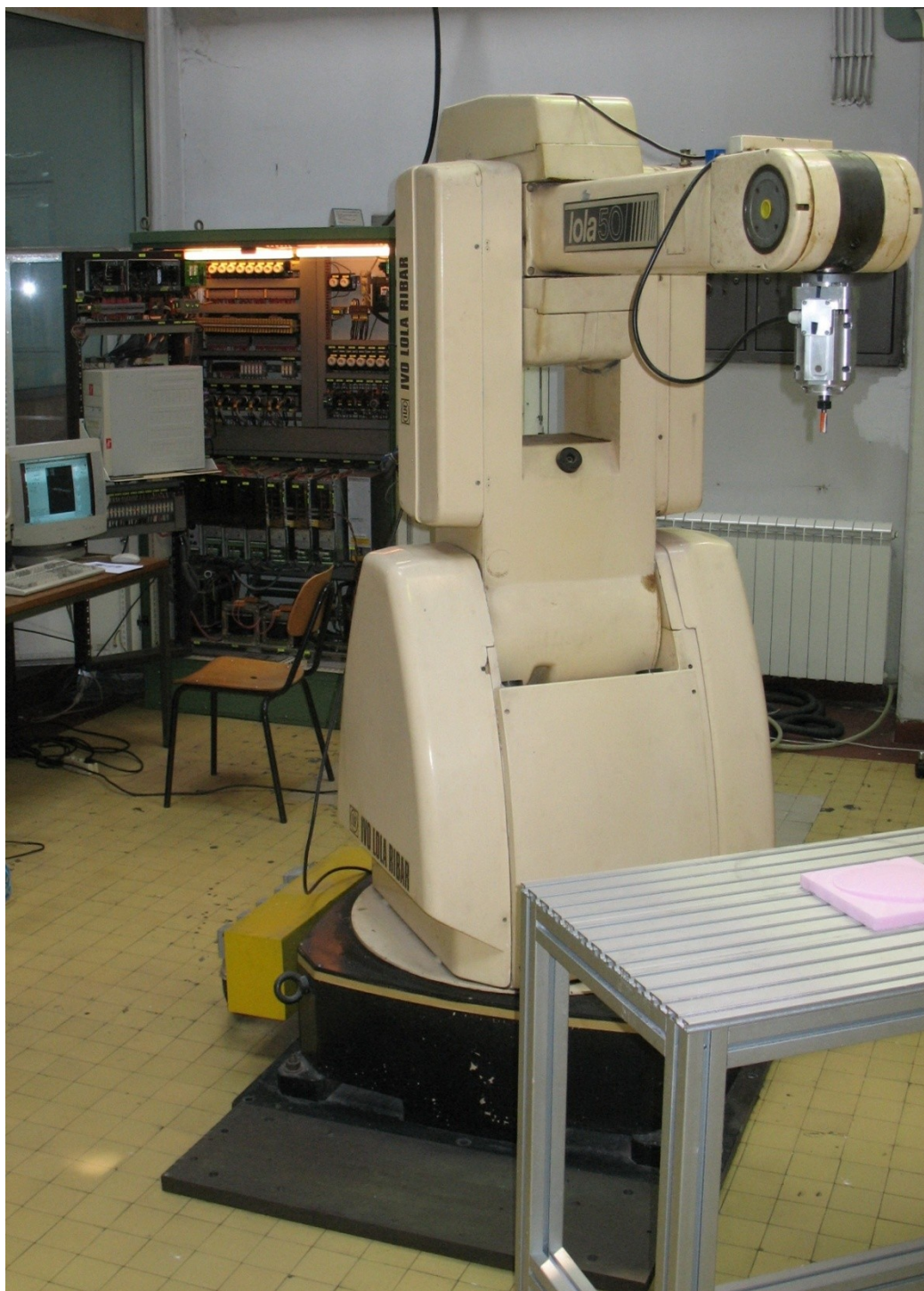


Слика 4.12 Начини реконфигурисања елементарне обрадне ћелије на бази робота вертикалне зглобне конфигурације са 6 степени слоборе

хоризонталне или вертикалне 5-осне машине алатке, нпр приликом преласка са конфигурације к3.1, слика 3.3, на конфигурацију к3.4, слика 3.6, или са конфигурације к3.1, слика 3.3, на конфигурацију к3.2, слика 3.4.

4.1.5 Реализација система управљања и програмирања на бази EMC2 софтвера

За потребе експерименталне верификације управљачког система на бази EMC2 софтвера на располагању је био робот вертикалне зглобне конфигурације Лола 50 са 6 степени слободe, слика 4.13. Управљачки систем робота Лола 50 је био заснован на робот контролеру предвиђеном за програмирање обучавањем и управљање PTP (енг. *Point-To-Point*) кретањима. Применом успостављене методологије развоја реконфигурабилног управљачког система на бази готових хардверских и софтверских система отворене архитектуре, реализован је управљачки систем коришћењем EMC2 софтвера, Linux оперативног система са *real-time* језгром, PC хардвера и интерфејс картица за улазно/излазне операције. Постојећи робот контролер је замењен новом управљачком јединицом која је повезана са постојећим електричним системима и аналогним погонима серво мотора, слика 4.13. PCI (*Peripheral Component Interconnect*) интерфејс ка матичној плочи PC рачунара је био један од главних критеријума за избор MOTENC Lite интерфејса за улазно/излазне операције, односно за спрегу са сензорима и погонима мотора. Две овакве картице, које су придодате основној рачунарској платформи, имају укупно осам аналогних излаза за везу са фреквенцијским регулаторима и осам квадратурних дигиталних улаза за везу са оптичким мерним системима (инкременталним енкодерима у овом случају). Ту је и мноштво дигиталних улаза и излаза за дигиталну интеракцију. Током експерименталне верификације, која је дата у поглављу 5, обрадна ћелија на бази робота Лола 50, са реализованим управљачким системом на бази EMC2 софтвера, је емулирала 5-осне вертикалне глодалице у конфигурацијама к3.1, слика 3.3, к3.3, слика 3.5, и к3.5, слика 3.7, као и 5-осне хоризонталне глодалице, за случајеве к3.2, слика 3.4, и к3.4, слика 3.6.



Слика 4.13 Елементарна обрадна ћелија на бази робота Лола 50 са реализованим управљачким системом применом софтвера *EMC2*

4.2 Развој реконфигурабилног управљања на бази софтверских алата и технологија отворене архитектуре и сопствених алгоритама управљања применом *OROCOS*-а

Општи приступ конфигурисању комерцијалних реконфигурабилних управљачких система отворене архитектуре, на бази развоја алгоритама управљања и одговарајућих софтверских компонената, изискује ангажовање обимних развојних ресурса. При томе се уобичајено спроводе следеће активности:

- Платформски независан развој потребних алгоритама и софтверских функција управљања и/или модификација постојећих;
- Избор рачунарске платформе одговарајућих перформанси;
- Избор оперативног система за рад у реалном времену у функцији изабране рачунарске платформе;
- Избор софтверских библиотека и компајлерских алата за имплементацију софтверских компонената управљања, као што је нпр. *OROCOS* који је коришћен у овој дисертацији.

Поставка методологије развоја реконфигурабилног управљања, на бази софтверских алата и технологија отворене архитектуре и сопствених алгоритама управљања применом *OROCOS*-а, у овој дисертацији обухвата:

- Развој потребних алгоритама и софтверских функција управљања и модификацију постојећих:
 - Развој компилатора *G*-кôда применом метода пројектовања структурираних програмских језика на објектно оријентисани начин;
 - Модификацију софтверских функција постојећег интерполатора путање [82], развијеног у Лола институту, који по перформансама, квалитету генерисане путање и могућностима превазилази слична готова решења у домену отворене архитектуре;
 - Имплементацију софтверских кинематичких функција на бази решења инверзне и директне кинематике, добијених применом генерализованог приступа кинематичког моделирања обрадних

ћелија на бази робота, приказаног у поглављу 3;

- Развој драјверских функција интерфејса за улазно/излазне операције;
- Развој *OROCOS* софтверских компонената са претходно развијеним софтверским функцијама управљања;
- Развој метода за конфигурисање и реконфигурисање управљачког система на бази развијених *OROCOS* софтверских компонената.

С обзиром на чињеницу да би комплетан развој оваквог реконфигурабилног управљачког система био велики задатак, који захтева обиман програмерски рад, у дисертацији је приказана имплементација основних софтверских компонената, коришћењем *OROCOS* софтверских алата.

G-кôд компилатор је у овој фази развијен на нивоу функционалних софтверских објеката. За комплетирање *G*-кôд *OROCOS* компоненте је неопходно уложити још развојних ресурса, због чега је за потребе експерименталне верификације конфигурисан управљачки систем кога поред модула интерполатора, кинематике и драјвера чини модул компилатора постојећег језика *L-IRL (Lola-Industrial Robot Language)*. *L-IRL* припада класи виших програмских језика и развијен је у Лола институту, према стандарду *DIN 66312 - Industrial Robot Language (IRL)*, за потребе програмирања и управљања индустријским роботима[83].

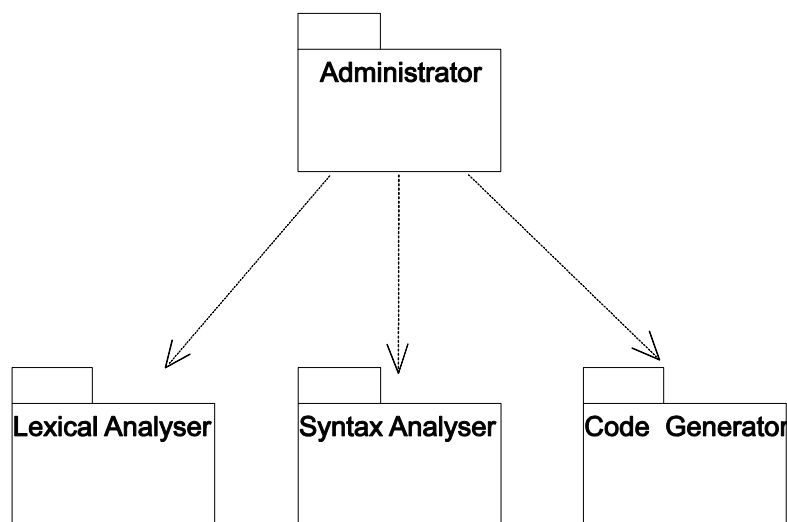
4.2.1 Развој алгоритама и софтверских функција управљања

У овом поглављу ће бити приказан развој потребних и модификација претходно развијених управљачких алгоритама и софтверских функција за управљање реконфигурабилним обрадним ћелијама на бази робота. Ови алгоритми управљања су развијени као платформски независни и могу се имплементирати као *real-time* софтверске функције и објекти на неком од објектно оријентисаних програмских језика, какав је нпр. *C++*, уз коришћење одговарајућих софтверских алата за распоређивање и синхронизацију у временском домену.

Компилятор G-кôда

Уобичајено је да се приликом реализације компилатора за структуриране програмске језике користи неки од генератора компилатора, што у многоне олакшава кодирање софтверског система. Због специфичности синтаксе и семантике G-кôда, компилатор предложен у овој дисертацији, реализован је мануелно, методом рекурзивног спуста (*recursive descent*) [84, 85] по свим правилима за реализацију компилатора виших, структурираних програмских језика. Имплементација компилатора G-кôда, *NC (Numerical Control)* компилатора, по методологији за реализацију компилатора виших програмских језика, пружа могућност проширења G-кôда инструкцијама и структурама карактеристичним за структуриране језике, као што су потпрограми, програмски блокови, структуриране петље, изрази и променљиве. Поменуте структуре нису дефинисане стандардом за имплементацију G-кôда, *RS274NGC*, односно (*The International Organization for Standardization*) *ISO 6983* или (*Deutsches Institut für Normung*) *DIN 66025*, али нуде низ погодности приликом ручног програмирања обраде.

Полазећи од генерализоване структуре компилатора, која укључује лексички и синтаксни анализатор и код генератор, слика 4.14, пројектовању компилатора претходи дефинисање *BNF (Backus-Naur Form)* нотације програмског језика, у овом случају G-кôда.



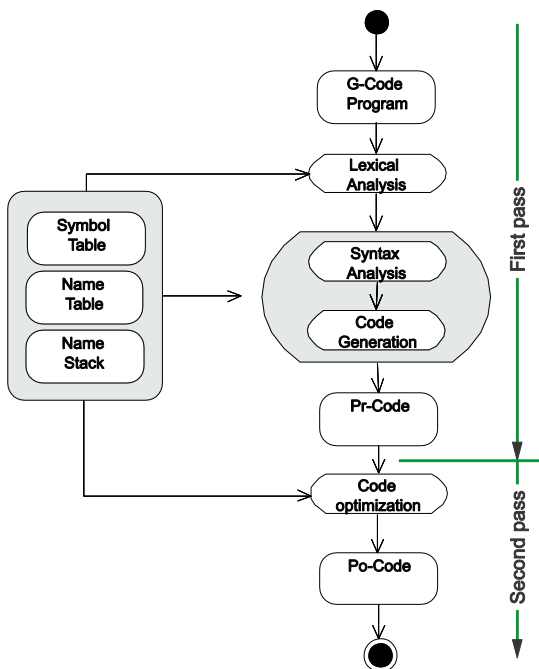
Слика 4.14 UML дијаграм софтверских модула *NC* компилатора

У табели 4.2 је дата *BNF* нотација за основни скуп G-кôд инструкција који је потребан и довољан за програмирање једне роботске ћелије за обраду.

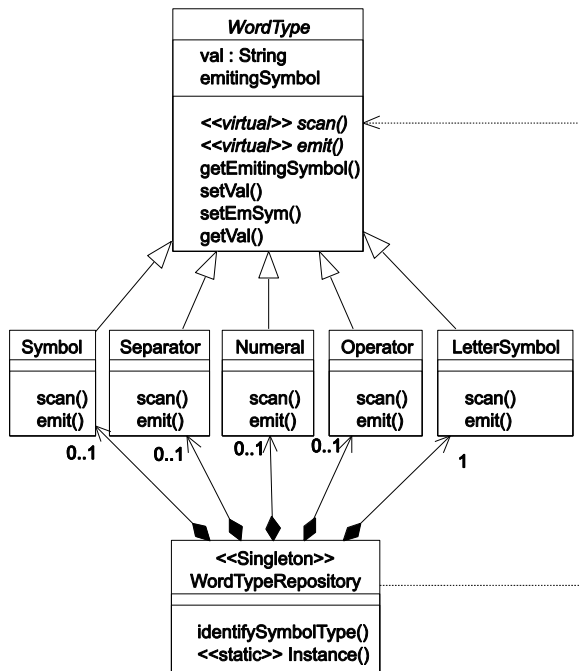
Табела 4.2 BNF нотација за основни скуп инструкција G-кода

BNF правила	Ограничења
Program = ProgramBorder StartProgram ProgramBody ProgramEnd ProgramBorder	
ProgramBorder = "%"	
StartProgram = "L" "INTEGER"	"INTEGER" треба да садржи тачно четири цифре
ProgramBody = Line {Line}	
ProgramEnd = "M30" "M02" "NEW_LINE"	M02 или M30 мора бити сам у линији
Line = "NEW_LINE" LineNumber [ExactStopInstruction StopInstruction InstructionQueue]	
LineNumber = "N" "INTEGER"	Бројеви линија морају бити у растућем редоследу
InstructionQueue = { GInstruction MInstruction Coordinates SCode SRCODE FCode HCode }	
ExactStopInstruction = G09 (G04 [PCode])	
StopInstruction = "M00"	
SCode = "S" "INTEGER"	SCode се јавља највише једном у истој линији; INTEGER у опсегу: 00 - 99999
HCode = "H" "INTEGER"	HCode се јавља највише једном у истој линији; INTEGER у опсегу: 00 - 99999
FCode = "F" "REAL"	FCode се јавља највише једном у истој линији; "REAL" је у формату 5.1
PCode = "P" "INTEGER"	PCode се јавља највише једном у истој линији;
Coordinate = ("X" "Y" "Z" "I" "J" "K" "R") SimpleExpression	
SimpleExpression = [Sign] Monom {AddOperatorAndMonom}	
Sign = "+" "-"	
Monom = Factor {MulOperatorAndFactor}	
AddOperatorAndMonom = AddOperator Monom	
MulOperatorAndFactor = MulOperator Factor	
Factor = SimpleExpressionWithBrackets "Real" Variable	
SimpleExpressionWithBrackets = "OPEN_BRACKET" SimpleExpression "CLOSED_BRACKET"	
GInstruction = P0Group P1Group P3Group P4Group P5Group P6Group P7Group P8Group	
P0Group = "G15" "G16"	
P1Group = "G00" "G01" "G02" "G03"	
P3Group = "G17" "G18" "G19"	
P4Group = "G20" "G21"	
P5Group = "G43" "G44"	
P6Group = "G53" "G54" "G55" "G56" "G57" "G58" "G59"	
P7Group = "G61" "G62" "G64"	
P8Group = "G90" "G91"	
MInstruction = "M03" "M04" "M05" "M07" "M07" "M08" "M09" "M10" "M11" "M60" "M61" "M80" "M81" "M98" "M99"	

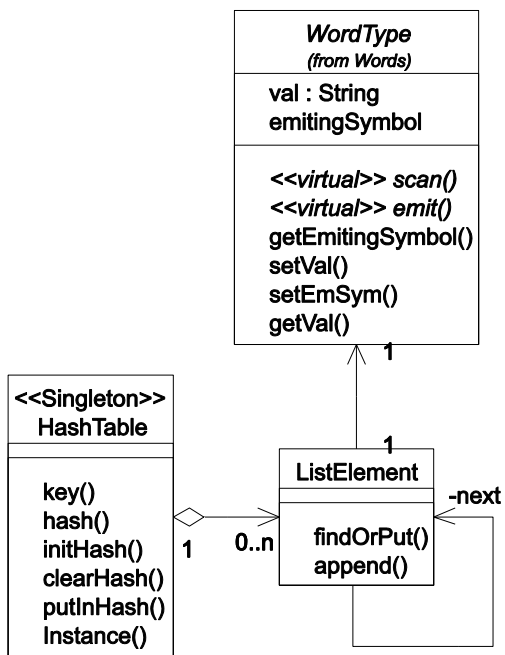
Компилација се обавља у два пролаза, слика 4.15. У првом пролазу софтверски модул *Lexical analyser*, слика 4.16, чита карактер по карактер из програмске *G*-кôд датотеке, формира речи и уз коришћење *Hash* алгоритма, слика 4.17, проналази их у табели *Symbol Table*, слика 4.18.



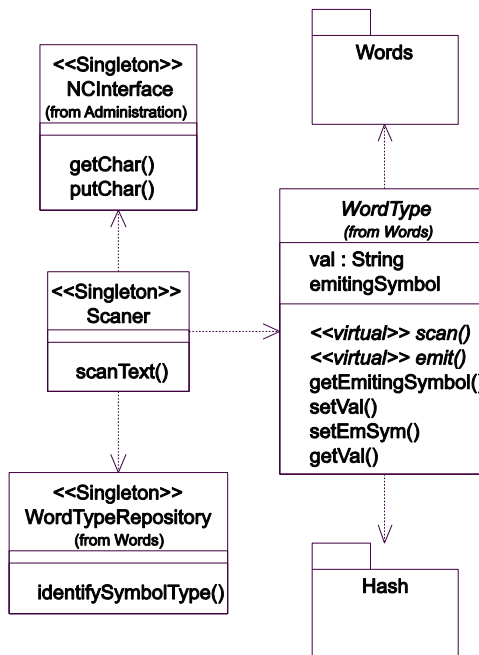
Слика 4.15 UML дијаграм активности NC компилатора



Слика 4.16. UML дијаграм класа *Lexical Analyser*-а

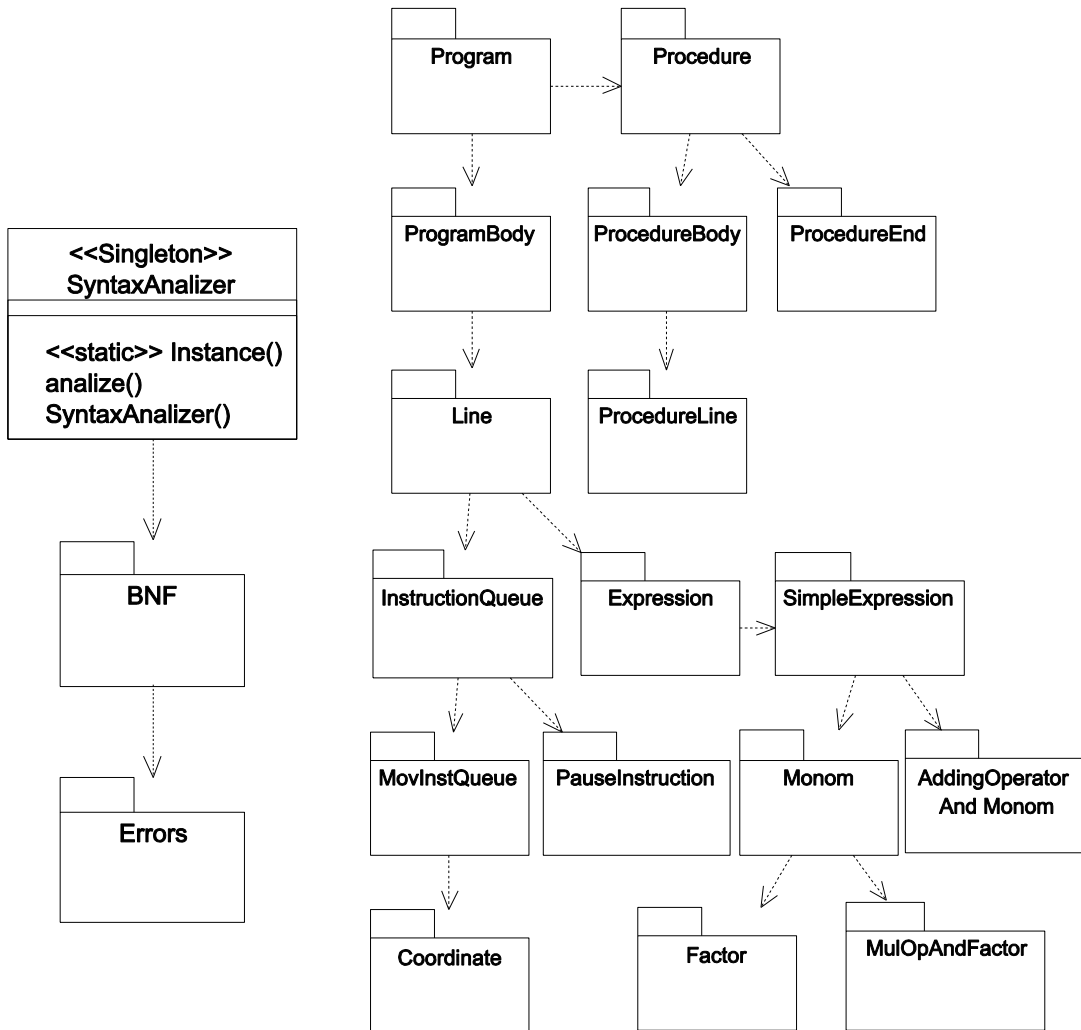


Слика 4.17 UML дијаграм *Hash*-а



Слика 4.18 UML дијаграм класа *Symbol Table*

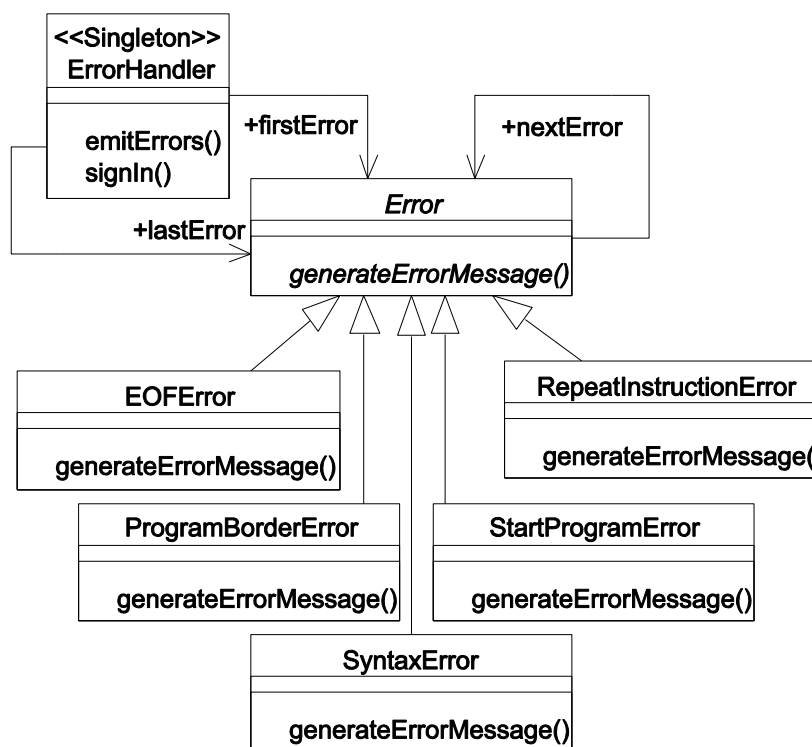
Приликом имплементације *Syntax Analyser*-а, слика 4.19, коришћена су претходно дефинисана *BNF* правила. Болдована имена из табеле 4.2 су постала имена *C++* класа које садрже алгоритме за проналажење одговарајућег блока или секвенце програма обраде, слика 4.20. Сви *BNF* објекти су инстанце класа изведених из апстрактне класе *BNFRule* са две апстрактне функције чланице: *public: checkRule()* и *private: expect()*.



Слика 4.19 UML дијаграм *Syntax Analyser*-а

Слика 4.20 UML дијаграм софтверских модула *BNF*-а

У изведеним класама, апстрактне функције су дефинисане у складу са захтеваном семантиком *BNF* правила. Након синтаксне и семантичке анализе, слика 4.15, генерише се међукод (*P*-кôд), или објекат класе *Error*, слика 4.21, који емитује поруку о грешци.



Слика 4.21 UML дијаграм класа *Error Handler*-а

У другом пролазу се решавају обраћања унапред и врши оптимизација кода, генерисаног у првом пролазу. Све активности током компилације координира синглтон објекат класе *Administrator*.

Рекурзивни спуст захтева задовољавање извесних ограничења у граматици језика. Граматика језика мора бити $LL(1)^3$ граматика. Осим тога, компилатор се мора реализовати на неком језику вишег нивоа, који дозвољава рекурзију.

Током корака компилације, слика 4.15, P -кôд пролази кроз неколико фаза. P_o -кôд је добијен оптимизацијом P_r -кôда и прилагођен тако да подржава извршавање програма обраде у реалном времену, наредбе кретања и модуларност. P -кôд је третиран као посебан језик, који има кодну синтаксу, којом су дефинисане кодне наредбе и кодне реченице. При пројектовању кодне синтаксе тежило се да кодне наредбе директно одговарају концептима и изворној синтакси G -кôда. P -кôд је бинарна датотека која садржи симболе дефинисане у следећој *enum* структури $C++$ програмског језика:

³ Силазно парсирање при коме се користи један *lookahead* токен за доношење одлука

```
enum NCSymbol
{
    NEW_LINE,
    PROGRAM_BORDER, START_PROCEDURE,
    G0, G1, G2, G3,
    G4, G9,
    G17, G18, G19,
    G22,
    G43, G44,
    G53, G54, G55, G56, G57, G58, G59,
    G61, G62, G64,
    G90, G91,
    G99,
    M0, M2, M3, M4, M5, M6, M7, M8, M9, M10, M11, M30, M60, M61, M80, M81, M98, M99,
    L, N, H, F, S, SR, P, X, Y, Z, A, B, C, I, J, K, R, V,
    ASSIGN, PLUS, MINUS, ADD, SUB, MUL, DIV,
    NUMERAL, REAL, INTEGER, SYMBOL, OPEN_BRACKET, CLOSED_BRACKET,
    UNKNOWN, UNKNOWN_COMMENT, UNKNOWN_LINE_NO,
    EMPTY
};
```

Овако дефинисана структура омогућава интерпретирање *P*-кôда у једном пролазу на *real-time* софтверском подсистему. Уз помоћ *case* програмске структуре *real-time* контролер идентификује симбол по симбол програма и попуњава структурну променљиву *command*. Тек након што је структура променљиве *command* попуњена, предузима се одговарајућа акција. Као пример, у табели 4.3 је приказан садржај променљиве *command* који је настао као резултат компилације једне *G*- кôд линије „N80 G2 G17 X0. Y-100. R-50“.

Табела 4.3 Пример компилације једне *G*- кôд линије

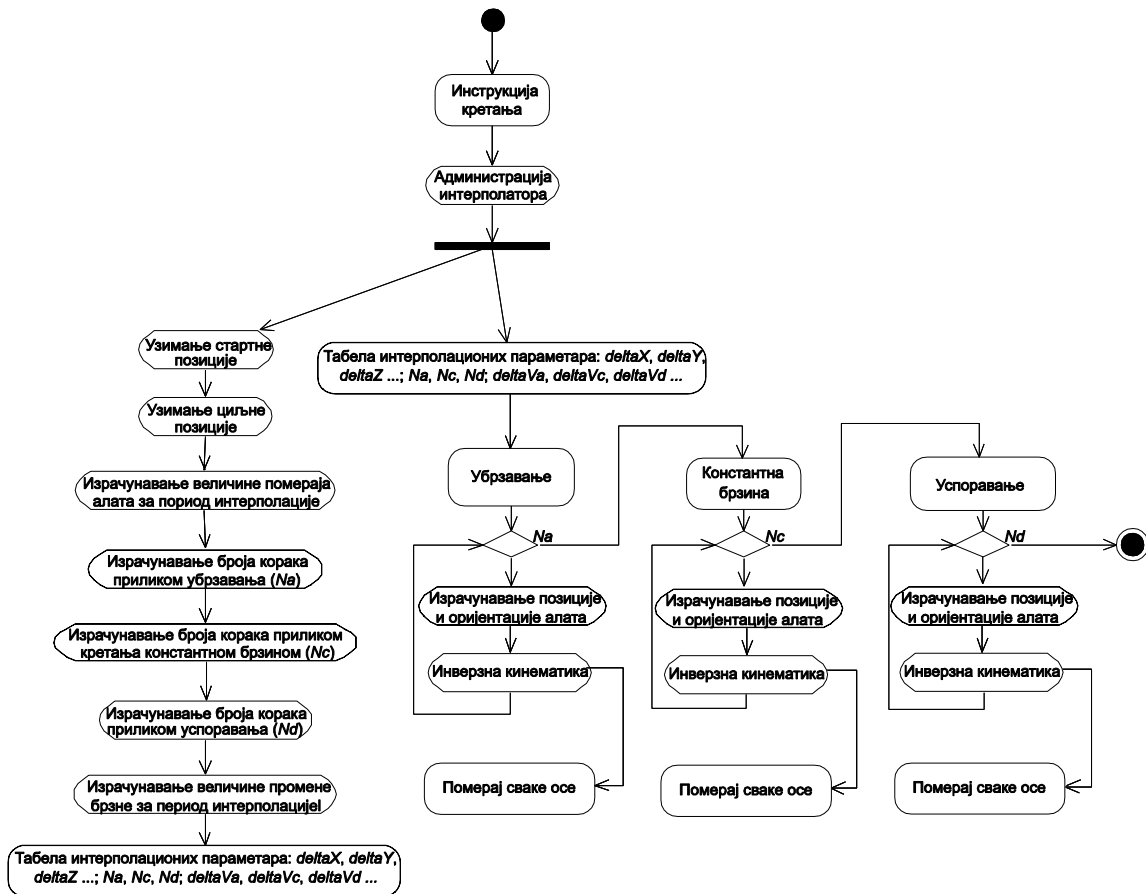
<i>G</i> - кôд	N80	G2	G17	X0.	Y-100.	R-50.
Симболичк и <i>P</i> - кôд	N 0x50	G2	G17	REAL 0 ASSIGN X	REAL ASSIGN 100 MINUS ASSIGN Y	REAL ASSIGN 50 MINUS ASSIGN R
<i>P</i> - кôд hex формат	<u>2F 00</u> <u>00 00</u> <u>50</u>	05	09	<u>48 00 00 00 00</u> <u>00 00 00 00 40</u> <u>35</u>	<u>48 40 59 00 00 00</u> <u>00 00 00 00 42 40</u> <u>36</u>	<u>48 40 49 00 00</u> <u>00 00 00 00 00</u> <u>42 40 3E</u>

Да би *P_o*-кôд био означен као валидан за извршавање на роботској ћелији за обраду, мора проћи проверу која укључује интерполацију инструкција кретања у *off-line* подсистему. *Off-line* подсистем постаје виртуелни *CNC* који извршава инструкције кретања генеришући путању, профиле брзина и убрзања алата, као и појединачних оса роботске ћелије, посредством кинематичког модула. Проверава се да ли су физичке осе у границама дефинисаног опсега кретања, брзина и убрзања. Посебне погодности током провере пружа могућност конфигурисања виртуелне роботске ћелије за симулацију програма у реалном времену. Ако током провере није емитована грешка међукод се означава исправним за извршење на роботској ћелији (*P_t*).

Интерполатор

Предложени управљачки систем интерполира путању користећи *Sampled-Data interpolation method* [86]. За разлику од интерполатора описаног у [86] код кога се интерполација врши са претпоставком да је брзина током програмиране путање константна, а затим се према потреби примењује *acceleration/deceleration* алгоритам, интерполатор коришћен за реализацију управљања у овој дисертацији генерише трапезни профил брзине за програмирану путању током интерполације.

Као што се на *UML* дијаграму активности са слика 4.22 може видети, коришћени интерполатор интерполира путању кроз активности које се извршавају у две фазе.



Слика 4.22 *UML* дијаграм активности општег интерполатора

У првој фази интерполације, израчунавају се интерполациони параметри појединих сегмената кретања. На пример, у случају петоосне обраде у конфигурацији (X, Y, Z, A, B) , израчунавају се следећи параметри:

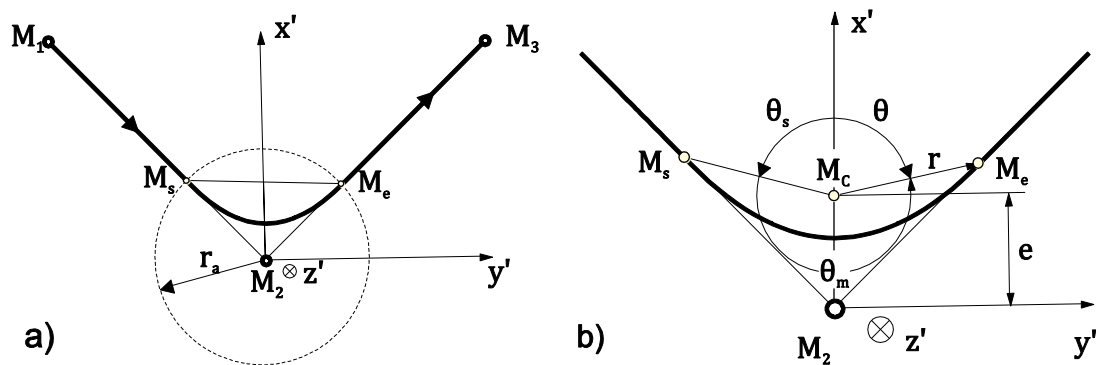
- $deltaX, deltaY, deltaZ, deltaA, deltaB$ - X, Y, Z, A, B помераји оса током једног

интерполационог периода;

- N_a, N_c, N_d - број циклуса убрзавања, кретања константном брзином и успоравања током једне инструкције кретања;
- $\Delta V_a, \Delta V_c, \Delta V_d$ - промене брзина током интерполационог периода.

У другој фази, за сваки период интерполације (Δt) се израчунава задати положај оса робота. При томе се најпре израчунава позиција алата у координатном систему емулиране машине $\{M\}$, слике 3.1 до 3.7, а затим се помоћу функција инверзне кинематике израчунавају положаји оса роботске ћелије за сваки интерполациони интервал. При томе се, уколико је то захтевано програмом обраде, међусобно повезују суседна кретања заглађивањем путање уз благу корекцију брзине.

Интеграторима система је дата могућност да дефинишу максимална убрзања оса за конкретну ћелију, као и начин повезивања два кретања у случајевима када се захтева континуална интерполација путање (*CPI-Continuous Path Interpolation*). Потребно је задати параметар који дефинише полупречник области апроксимативног кретања (r_a), односно полупречник области у којој алат одступа од програмиране путање у циљу смањења инерцијалних сила и одржања програмиране брзине обраде, слика 4.23.



Слика 4.23 Апроксимативно кретање

Координате тачака M_s и M_e се координатном систему емулиране машине $\{M\}$, добијају као:

$$[x_s, y_s, z_s]^T = [x_2, y_2, z_2]^T + \frac{r_a}{s_{ms}} [x_1 - x_2, y_1 - y_2, z_1 - z_2]^T \quad (4.1)$$

$$[x_e, y_e, z_e]^T = [x_2, y_2, z_2]^T + \frac{r_a}{s_{me}} [x_3 - x_2, y_3 - y_2, z_3 - z_2]^T \quad (4.2)$$

при чему су s_{ms} и s_{me} дужине M_1M_2 и M_3M_2 . Једначина параболе се у $x'y'$ равни, која се поклапа равни у којој је врши кретање алата, може изразити као:

$$x' = a_2 y'^2 + a_0 \quad (4.3)$$

Из диференцијала ове једначине у тачки M_s :

$$\frac{dx'(y'_s)}{dy'_s} = 2a_2 y'_s \quad (4.4)$$

добиају се следећи параметри:

$$a_2 = \frac{x'_s}{2y'^2_s} \quad (4.5)$$

$$a_0 = \frac{x'_s}{2} \quad (4.6)$$

На основу ових параметара, једначина параболе се може изразити на следећи начин:

$$y'^2 = 2 \frac{y'_s}{x'_s} \left(x' - \frac{x'_s}{2} \right) \quad (4.7)$$

или у параметарском облику као:

$$x' = e + r \cos \theta \quad (4.8)$$

$$y' = r \sin \theta \quad (4.9)$$

при чему су:

$$r = \frac{y'^2_s}{x'_s(1 - \cos \theta)} \quad (4.10)$$

$$e = \frac{x'^2_s + y'^2_s}{2x'_s} \quad (4.11)$$

$$\theta \in [\theta_s, \theta_e], \theta_s = \text{atan } 2 \left(\frac{y'_s}{(x'_s - e)} \right) \quad (4.12)$$

Угао лука параболе је према слици 4.23б:

$$\theta_m = 2(\pi - \theta_s), \text{ ако је } \theta_m > 0, \text{ односно } \theta_m = 2(\pi + \theta_s), \text{ ако је } \theta_m < 0 \quad (4.13)$$

Овде је:

$$\theta_e = 2\pi - \theta_s, \text{ ако је } \theta_s > 0, \text{ односно } \theta_e = -(2\pi - \theta_s), \text{ ако је } \theta_s < 0 \quad (4.14)$$

Интерполација путање током апроксимативног кретања се може вршити као да се ради о додатом кретању или деформисаном линеарном кретању. Предложени интерполатор је развијен усвајањем концепта који подразумева деформацију постојећих програмираних кретања у апроксимативној области. При томе је обезбеђено тангентно напуштање програмираног кретања и тангентно враћање на следеће програмирано кретање. Током кретања у

апроксимативној области свака интерполациона тачка по програмираној путањи се у реалном времену замењује одговарајућом тачком на параболу у апроксимативној области. Број сегмената интерполације током кретања у апроксимативној области остаје исти као да се кретање врши у режиму без заглађивања путање. Овакав приступ доводи до благе линеарне промене брзине у току заглађивања али смањује број неопходних израчунавања у односу алгоритам који подразумева додавање нове путање у апроксимативној области [82].

Из скупа функција за интерполацију путање које су садржане у интерполатору развијеном у Лола институт [82], за потребе имплементације реконфигурабилног управљачког система роботских хелија за обраду, коришћене су следеће:

- *Planer* - администраторска функција интерполатора задужена за уланчавање инструкција кретања у циљу повезивања путање, која позива функцију:
 - *Movex* - администраторска функција интерполатора за селекцију функције за прорачун интерполационих параметара и одговарајуће функције за интерполацију путање:
 - *cp_par* - функција интерполатора која врши селекцију једне од функција за прорачун интерполационих параметара за кретања по аналитички описивим путањама:
 - *get_lin_var* - прорачун интерполационих параметара за кретање по линеарној путањи са променљивом оријентацијом алата;
 - *get_cir_var* - прорачун интерполационих параметара за кретање по кружној путањи са променљивом оријентацијом алата;
 - *lin_tin_var* - функција интерполатора за линеарну интерполацију путање са променљивом оријентацијом алата;
 - *cir_tin_var* - функција интерполатора за кружну интерполацију путање са променљивом оријентацијом алата.

Кинематичке функције

У поглављу 3 су применом развијеног генерализованог приступа кинематичког моделирања решени инверзни и директни кинематички

проблеми елементарне обрадне ћелије на бази робота са 5 и 6 степени слободе, који емулирају 5-осне машине алатке различитих конфигурација. У конфигурацијама к1.1 и к2.1 робота са 5 степени слободе, посредством функција инверзне и директне кинематике остварено је директно пресликавање спољашњих координата 5-осне машине алатке у унутрашње координате робота са 5 степени слободе и обрнуто.

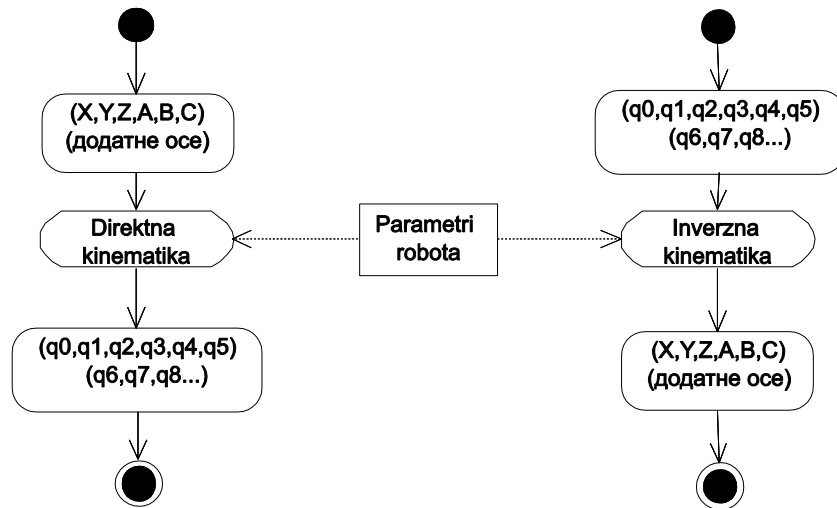
У случајевима када је робот са 6 степени слободе моделиран тако да емулира 5-осну машину алатку, у циљу једнозначног пресликавања координата коришћен је један од следећа два приступа:

- Блокирана је једна од оса робота, односно унутрашња координата је држана константном (нпр. $\theta_6 = 0^\circ$ или $\theta_4 = 0^\circ$ или $\theta_4 = \pm 90^\circ$), као што је то показано у поглављу 3 за конфигурације к3.1, к3.2 и к3.5;
- Спољашња координата, која не фигурише у 5-осној обради емулиране машине алатке, држана је константном (нпр. $C = 0^\circ$ или $B = 0^\circ$), као што је то показано у поглављу 3 за конфигурације к3.3 и к3.4.

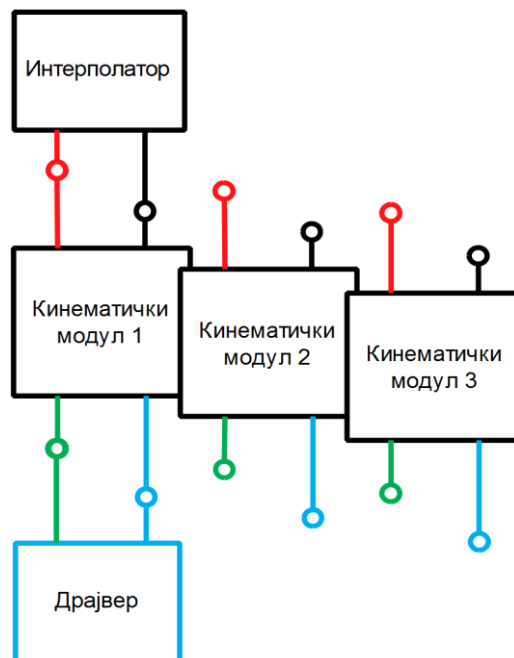
С обзиром на чињеницу да оптимизација кода управљачког система у смислу заузећа меморијског простора није била предмет ове дисертације, без обзира на истоветност појединих решења инверзне и директне кинематике емулираних 5-осних машина алатки, њихове имплементације су садржане у одвојеним софтверским функцијама.

У општем случају, два функционална блока са интегрисаним функцијама инверзне и директне кинематике и табела кинематичких параметара роботске ћелије за обраду, садржана у датотеци заглављу, су основни елементи кинематичког модула, слика 4.24.

Једну обрадну ћелију на бази робота у управљачком систему представља скуп кинематичких модула који моделирају изабране кинематичке конфигурације дате обрадне ћелије. У одабраној радној конфигурацији, одговарајући кинематички модул, посредством униформног апликативног интерфејса, остварује везе са осталим функционалним модулима у систему, слика 4.25.



Слика 4.24 UML дијаграм активности општег кинематичког модула

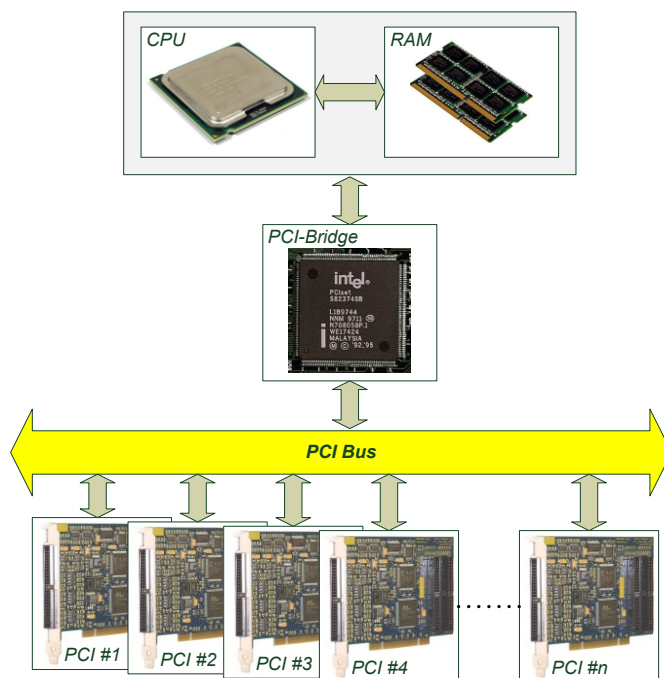


Слика 4.25 Униформни интерфејс кинематичких модула

Драјверске функције интерфејса за улазно/излазне операције

Отвореност архитектуре, као предуслов реконфигурабилности управљачког система, намеће употребу оперативног система отворене архитектуре и комерцијално расположивог хардвера (*COTS-Commercially Available Off-The-Shelf*). Предложени реконфигурабилни управљачки систем се заснива на *PC* архитектури и одговарајућим улазно/излазним (*I/O-Input/Output*) картицама. *PC* архитектура подразумева рачунарски систем

који на матичној плочи интегрише микропроцесорско окружење са одговарајућим магистралама података, као што је *PCI*, посредством којих се повезују одговарајући хардверски интерфејси у систем, слика 4.26.



Слика 4.26 Архитектура *PCI* подсистема

Хардверске компоненте засноване на *PCI* вези са матичном плочом су протекле деценије потврдиле своју поузданост и перформансе, дајући интеграторима система могућност да по ниској цени бирају између великог броја улазно/излазних картица и процесора високих перформанси. Коришћење управљачког хардвера на *PC* основи омогућава да се највећи део развоја, конфигурисања и дебаговања управљачког софтвера обави на класичним десктоп или лаптоп рачунарима, што знатно скраћује пут до коначног софтверског производа.

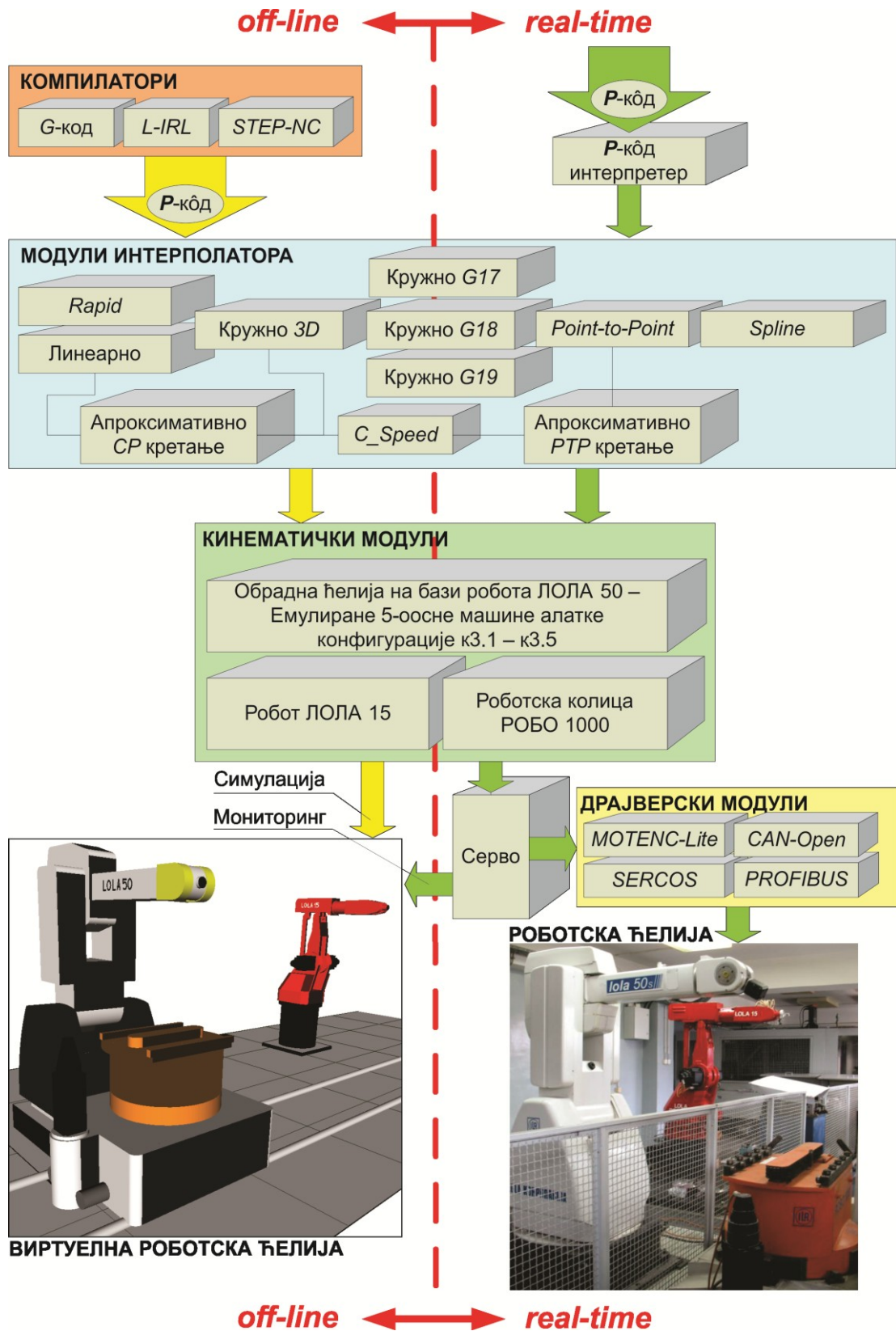
У општем смислу, драјвер неког периферијског уређаја представља софтверску компоненту оперативног система. Међутим, уколико се жели већа контрола над функционисањем периферних уређаја у реалном времену, драјвер се интегрише са конкретном *real-time* апликацијом. Током развоја управљачког система, приказаног у овој дисертацији, драјвери интерфејс картица за улазно/излазне операције су развијени и имплементирани у софтверску компоненту која је саставни део система, што је показано у наредним поглављима.

4.2.2 Поставка платформски независне структуре реконфигурабилног управљачког система

Предложен управљачки систем се састоји из *real-time* софтверског подсистема који је задужен за управљање роботском ћелијом и *off-line* подсистема за писање и едитовање програма, проверу синтаксе и семантике кода, и симулацију програма обраде у складу са кинематиком роботске ћелије за обраду. Као неизоставни корак током програмирања роботских ћелија за обраду, *Off-line* подсистем генерише међукод (*P-code*) који садржи кондензовани скуп инструкција, укључујући интерполационе параметре израчунате током процеса симулације. Извршавање тако добијеног међукода на *real-time* подсистему елиминише потребу за временски захтевним *look-ahead* алгоритмом и пружа могућност коришћења јефтиније хардверске платформе са нижим перформансама. Поменути приступ обезбеђује *real-time* детерминизам приликом читавања и обраде података са мерних система, интерполацију програмиране путање као и израчунавање инверзне и директне кинематике у сваком интерполационом интервалу. При томе је, на савременим и брзим хардверским платформама, временска редунданса више него довољна за обраду информација потребних за сервисирање захтева са корисничког интерфејса као и за комуникацију са удаљеним надзорним системима (виртуелним роботским ћелијама и сл.).

Захтев за реконфигурабилношћу намеће слојевиту, хијерархијску расподелу софтверских модула управљачког система. Хијерархијска, слојевита структура подразумева груписање модула према функционалној припадности као и према могућности повезивања са модулима суседних хијерархијских нивоа. Слика 4.27 даје генерички приказ комплетне хијерархијске структуре реконфигурабилног управљачког система који поред управљања роботском ћелијом у обради, нуди и конвенционалан начин програмирања работа.

Архитектура реконфигурабилног управљачког система треба да омогући правилан избор модула, који припадају одговарајућим хијерархијским нивоима, и њихову интеграцију у управљачки систем конфигуриране роботске ћелије за обраду. Да би то било задовољено, реконфигурабилни



Слика 4.27 Генеричка хијерархијска структура предложеног реконфигурабилног управљачког система

управљачки систем треба да буде пројектован у складу са следећим претпоставкама:

- Пројектовање софтвера управљачког система треба да подразумева објектно-оријентисани приступ (*OOP-Object Oriented Programming*) уз коришћење *UML*-а [81]. Поменути приступ обезбеђује одговарајуће документовање софтвера и поједностављује пројектовање;
- Управљачки систем треба да садржи потребан и довољан број софтверских модула на свим хијерархијским нивоима, као и методологију којом се дефинишу могуће управљачке конфигурације;
- Изван софтверске хијерархијске лествице управљања мора постојати систем који омогућава функционалан тест управљачке конфигурације у виду одговарајућих визуелних алата, као што је виртуелна роботска ћелија за обраду.

Комбиновање модула у оквиру хијерархијских нивоа даје могућност избора оптималне софтверске конфигурације за конфигурисану роботску ћелију и захтевану обраду. Тако је избор модула интерполатора повезан са конкретном апликацијом, односно обрадом која ће се извршавати, док је избор кинематичких модула у вези са кинематичком конфигурацијом роботске ћелије. Модуле компилатора бира корисник у складу са својим могућностима и потребама програмирања. Уколико се роботска ћелија програмира за обраду посредством *CAD/CAM* система, избор је усмерен на *G*-кôд или *STEP-NC* [87, 88] компилатор. За програмирање манипулативних радњи предвиђен *L-IRL* компилатор.

Два софтверска подсистема, *off-line* и *real-time*, на два хијерархијска нивоа имају идентичне софтверске модуле, слика 4.27, што омогућава комбиновање истих, ако се извршавају на истој хардверској платформи. Реч је о софтверским модулима који се налазе на интерполаторском и кинематичком хијерархијском нивоу.

***OROCOS* софтверски алати за примену у роботици**

OROCOS је настао као резултат европског пројекта покренутог 2001. године на белгијском Католичком универзитету у Лувену, уз учешће Лабораторије за анализу и архитектуру система (*CNRS/LAAS*) из Француске и

шведске лабораторије *KTH (Kungl Tekniska Högskolan)*. Поред ових институција на развоју и дизајну *OROCOS*-а учествују и многе друге лабораторије и институти. Основни циљ пројекта је развој модуларног софтверског окружења (енг. *framework*) отворене архитектуре (*open-source*) које пружа функционалну основу за имплементацију управљачких система робота.

OROCOS представља скуп оригиналних софтверских библиотека и алата, али укључује и друге софтверске системе који су доступни у домену отворене архитектуре. Даље ће бити приказани софтверски алати и библиотеке *OROCOS*-а, који су омогућили реализацију реконфигурабилног управљачког система роботских ћелија за обраду у овој дисертацији, при чему су коришћена документа доступна на интернет страници *OROCOS* пројекта [79].

OROCOS је реализован као софтверски систем, независан од платформе и апликације коју имплементира, са следећим карактеристикама:

- Отвореност и доступност софтверског система за проучавање, употребу и модификације;
- Модуларност и флексибилност која омогућава програмерима да креирају функционалне компоненте, без потребе за детаљним сагледавањем суштине целокупног софтверског система;
- Независност од комерцијалних произвођача робота, али и компатибилност са роботским системима неколико европских произвођача;

Отворена архитектура и стандарди у програмским језицима, форматима података, комуникационим протоколима и писању документације је од велике важности за слободно коришћење и кооперативни развој софтвера. Једино пројекти отворене архитектуре могу да гарантују оптималну употребу и поштовање отворених стандарда, док су произвођачи комерцијалног софтвера често у искушењу да уводе сопствене стандарде и протоколе због конкуренције на тржишту. Најважнији стандарди и софтверски системи отворене архитектуре које *OROCOS* подржава су:

- *CORBA IDL*, за уграђивање објеката. *IIOP (Internet Inter-Orb Protocol)*, обезбеђује интероперабилност различитих *CORBA* имплементација.

Поред тога, *CORBA* има *real-time* екстензију у виду ТАО (*The Ace Orb*) имплементације отворене архитектуре, која је веома битна за управљање роботима;

- *XML (eXtensible Markup Language)* структурирани језик текстуалних докумената, за конфигурисање система и датотека са подацима.
- *Modelica*, објектно оријентисани језик за моделовање модуларних комплексних система. Већина комерцијалних пакета за симулацију динамичких система подржавају овај формат моделирања.
- *DocBook* и *LATEX*, платформски независни системи отворене архитектуре за писање софтверске документације. Могу бити преведени на различите формате и као такви су погодни за обраду документације *Linux* софтверских пројеката. *DocBook* је погодан за писање корисничких упутстава, док је *LATEX* погодан за писање математичких формула.

Софтверски систем *OROCOS*-а се групише у софтверске компоненте (модуле) или софтверске библиотеке. Компонента у овом контексту означава софтверски објекат (*CORBA - Common Object Request Broker Architecture*, *DCOM - Distributed Component Object Model*), који може бити динамички додат и/или уклоњен из система, са функционалним језгром које је уоквирено одговарајућим неутралним апликативним интерфејсом, независним од коришћеног програмског језика (као што је *CORBA IDL-Interface Description Language*). У структури *OROCOS*-а се издвајају три основне групе софтверских модула:

1. *Модули за подршку*. То су софтверски модули опште намене неопходни за изградњу система за управљање роботима, као што су:
 - универзалне софтверске библиотеке,
 - софтверски алати за конфигурисање компонената,
 - модули за спрегу са оперативним системима за рад у реалном времену,
 - софтверски механизми за међупроцесну комуникацију и синхронизацију,
 - алати за писање документације...

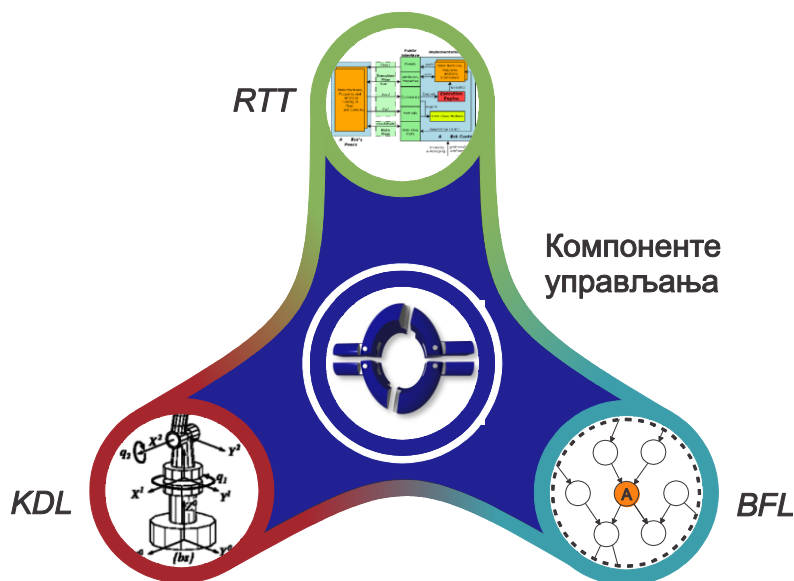
2. *Роботски модули*. То су софтверски модули који се ослањају на један или више модула за подршку и садрже елементарне роботске алгоритме као што су:

- нумерички алгоритми за израчунавање кинематике серијских кинематичких механизма,
- серво контролери,
- естиматоре за Бајесове и неуралне мреже,
- елементарни интерполатори путање...

3. *OROCOS Компоненте*. То су *CORBA* објекти са својим *IDL* описом који садрже модуле два претходна типа. Представљају градивне блокове помоћу којих корисници дизајнирају свој систем за управљање роботима.

OROCOS софтверски систем чине следеће софтверске библиотеке, слика 4.28:

- *RTT (Real-Time Toolkit)*, која сама по себи не представља апликацију већ обезбеђује инфраструктуру и функционалности за развој управљачких апликација на језику *C++*. Нагласак је на *real-time, on-line* интерактивним апликацијама базираним на компонентама. *RTT* се ослања на *real-time* кернел матичног оперативног система и енкапсулира га својим *real-time* апликативним интерфејсом.



Слика 4.28 Градивне софтверске компоненте *OROCOS*-а [79]

- *OCL (OROCOS Component Library)* обезбеђује библиотеку са ограниченим бројем компонената већ спремних за употребу. Неке компоненте могу да послуже као пример, а неке имплементирају веома комплексне алгоритме уз коришћење *RTT*-а;
- *KDL (Kinematics and Dynamics Library)* је *C++* библиотека која омогућава нумеричко израчунавање кинематике серијских механизма у реалном времену.
- *BFL (Bayesian Filtering Library)* обезбеђује апликационо независни оквир за реализацију динамичких Бајесових мрежа, нпр. рекурзивну обраду и естимацију алгоритама заснованих на Бајесовој теореме (Калманови филтри, итд.).

За конфигурисање управљачког система, *OROCOS* не намеће специјалне захтеве за употребу програмских језика приликом пројектовања софтвера. Може се користити *C* за *real-time* управљање, *C++* за нумеричке алгоритме, неки од скрипт програмских језика за графичке интерфејсе или роботске програме итд. Комуникацију између модула реализованих различитим програмским језицима обезбеђује *CORBA*. Шта више, усвајање *CORBA*-е као стандарда за комуникацију између модула пружа могућност њихове дистрибуције и извршавања у реалном времену на више умрежених хардверских платформи.

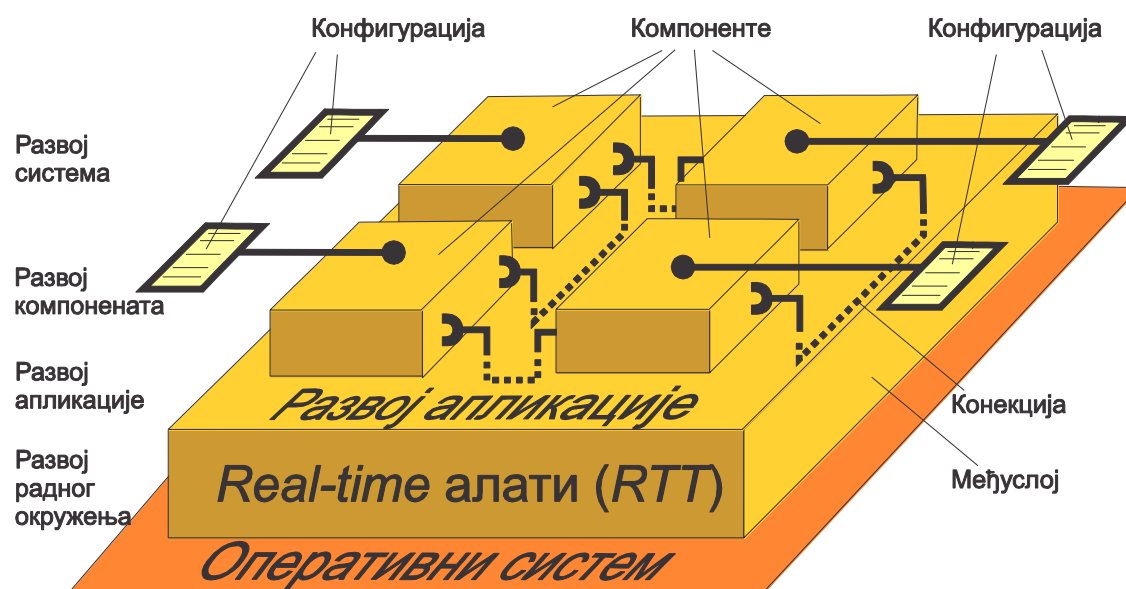
Готово сви модули *OROCOS*-а су на располагању и у другим *open-source* пројектима. На исти начин, неки од других *open-source* пројеката се могу уградити у *OROCOS*:

- *Octave*, веома богата нумеричка библиотека са *ODE (Ordinary Differential Equation)* и *DAE (Differential Algebraic Equation)* рачунарским алатима, која је компатибилна са *Matlab* пакетом;
- *RTLinux* или *RTAI*;
- *Xenomai*;
- *COMEDI*;
- Различити пакети отвореног формата за *3D* визуелизацију, као што су *VRML (Virtual Reality Modeling Language)*, *Java 3D*, *OpenGL*, *OpenInventor*, *Coin3D*, итд.

Компоненте су основне софтверске јединице *OROCOS*-а које обављају једну или више специфицираних активности. Процес конфигурисања управљачког система се састоји у повезивању *OROCOS* компонената у специфичну мрежну апликацију. Компоненте могу бити већ дефинисане или се могу развити сопствене, коришћењем *RTT* библиотеке или неке друге (са кинематиком, динамиком или визијом). Везе између компонената се дефинишу посредством *XML* заглавља (својстава), а успостављају преко команди/метода (енг. *command/method*) и/или токова података (енг. *data ports*) апликативног интерфејса.

Поред дефинисања механизма комуникације између компонената, *OROCOS* пружа могућност дефинисање хијерархијске машине стања у циљу распоређивања и секвенцирања рада компонената. То омогућава дефинисање специфичне логике корисничке апликације.

RTT је основна библиотека *OROCOS*-а, посредством које се остварује спрега корисничке апликације са *real-time* оперативним системом, слика 4.29. *Real-time* алати, које поседује ова библиотека, омогућавају изградњу, дистрибуцију и конфигурисање компонената апликације, као и њихову међусобну комуникацију у реалном времену.

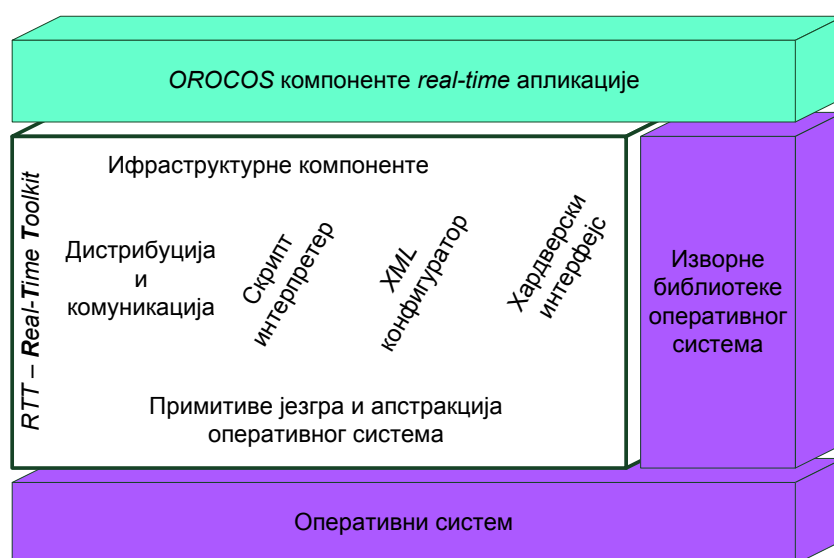


Слика 4.29 *OROCOS* окружење за развој *real-time* апликација

Интерно хијерархијски структурисан, *RTT* се својим хардверски оријентисаним слојем ослања на оперативни систем и улазно/излазне уређаје, слика 4.30.

Дизајн на бази *RTT*-а даје апликативним компонентама следеће могућности, слика 4.30:

- Реконфигурабилност компоненте посредством *XML* текстуалних датотека;
- Доступност апликативног интерфејса компоненте посредством мреже;
- Могућност задавања текстуалних команди посредством скрипт интерпретера садржаног у апликативном интерфејсу компоненте.



Слика 4.30 Структура *RTT*-а

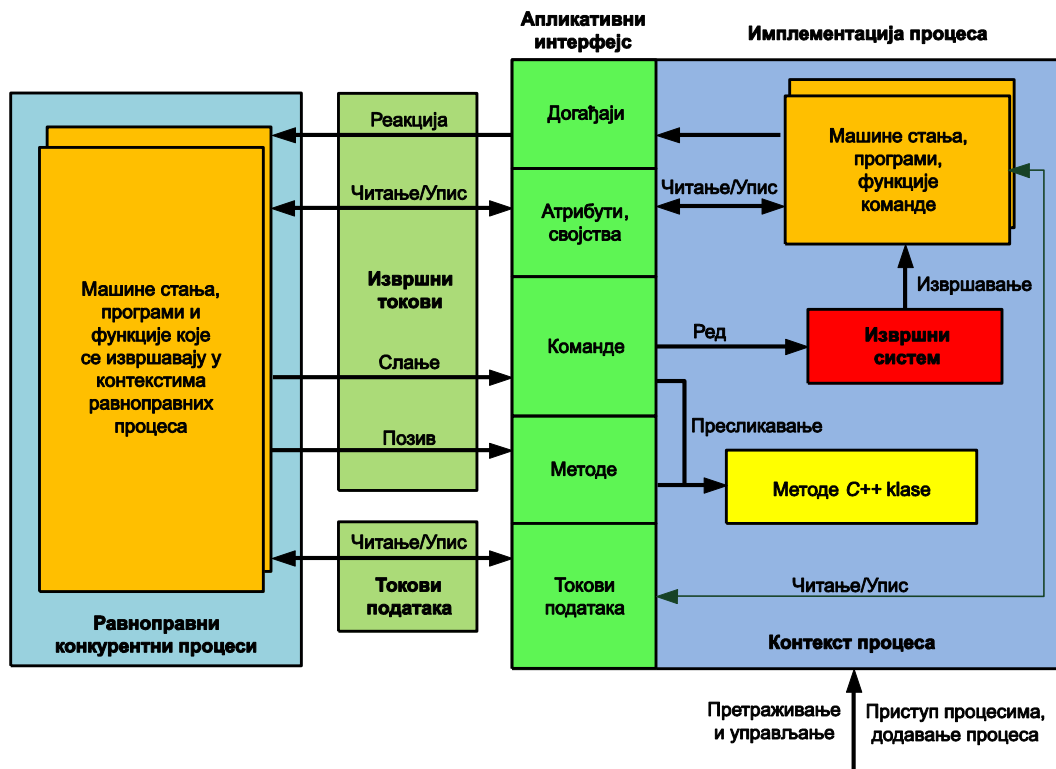
Софтверске компоненте реализоване применом *RTT* библиотека, имају униформни интерфејс за међусобно повезивање и изградњу управљачког софтвера за рад у реалном времену. Приликом конфигурисања система за рад у реалном времену, за сваку компоненту се мора дефинисати једна од следећих активности:

- Апериодична активност, која подразумева да се компонента активира сагласно неком спољном апериодичном процесу (*NonPeriodicActivity*);
- Периодична активност секвенцирана од стране интерног часовника реалног времена (*PeriodicActivity*);
- Секвенцијална активност у складу и са правилима секвенцијалних процеса у систему (*SequentialActivity*);

- Активност иницирана од стране друге компоненте у систему (*SlaveActivity*).

Свака *OROCOS* апликативна компонента наслеђује јавни интерфејс своје основне класе *TaskContex*, који садржи примитиве за међусобну интеракцију компонената, слика 4.31:

- Догађаје (енг. *Events*) - Догађаји омогућавају да се одговарајуће функције изврше након што је детектована промена (догађај) у систему (на пример: пристигла нова информација са сензора, корисник је покренуо нову акцију,...);
- Методе (енг. *Methods*) - Методе су функције које се позивају од стране других компонената и које се извршавају у контексту позивајућег таска. Методе имају структуру класичних *C* функција, али се могу писати и у скрипт језику и позивати преко мреже. Позив методе јесте у реалном времену, али није безбедан (енг. *thread-safe*), па је потребно да се одговарајући део кôда заштити *OROCOS Mutex* објектом;
- Команде (енг. *Commands*) - Команде су функције чије се извршавање иницира од стране других таскова, а које се извршавају у контексту



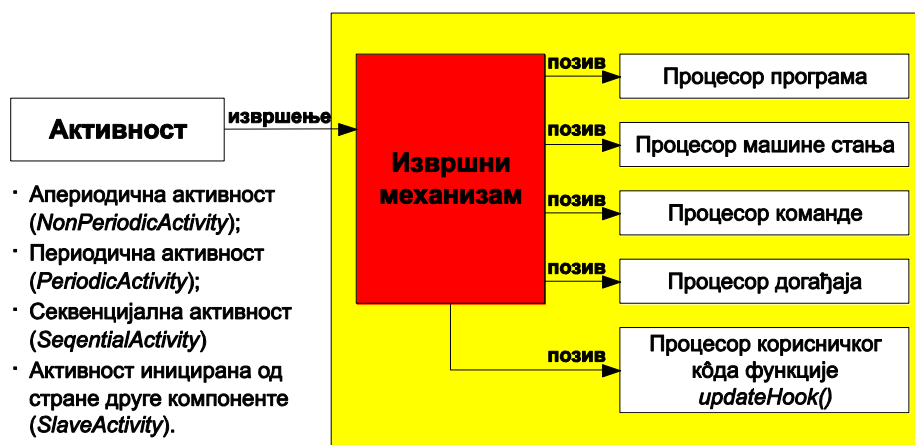
Слика 4.31 Интерфејс *OROCOS* апликативне компоненте (процеса)

матичне компоненте. Начин имплементације команде омогућава да се у сваком тренутку прати статус извршавања;

- Својства (енг. *Properties*) - Својства представљају параметре система, садржане у *XML* датотеци, који се могу изменити у току извршавања програма;
- Токове података (енг. *Data Ports*) - Токови података су механизам за безбедан (енг. *thread-safe*) транспорт података између компонената, при чему портови могу бити баферисани или небаферисани. Компонента може испитивати портове како би детектовала промену и прочитала податке или се може тригеровати (будити) када подаци пристигну;

Као што се са слике 4.31 може видети, градивни механизми за имплементацију функција *TaskContext* класе у реалном времену су:

- Извршни механизам (енг. *Execution Engine*) – Извршни механизам класе *TaskContext* обезбеђује оквир за извршавање програма, машина стања, команди, функција покренутих догађајем и корисничког кода у реалном времену. Извршни механизам компоненте се везује за одговарајући тип активности пре покретања, слика 4.32. На тај начин активне компоненте постају *real-time* нити са додељеним приоритетом између 0 и 99;
- Механизам за скриптовање (*OROCOS Scripting*) - Механизам за скриптовање омогућава да се програми и коначне машине стања, написани на скрипт језику (енг. *real-time scripting language*), преведу у



Слика 4.32 Активности које покрећу *TaskContext*

реалном времену у форму погодну за извршавање од стране извршног механизма.

Основне компоненте *OCL*-а које морају да уђу у састав управљачког система на бази *OROCOS*-а су:

- *TaskBrowser*;
- *DeploymentComponent*;

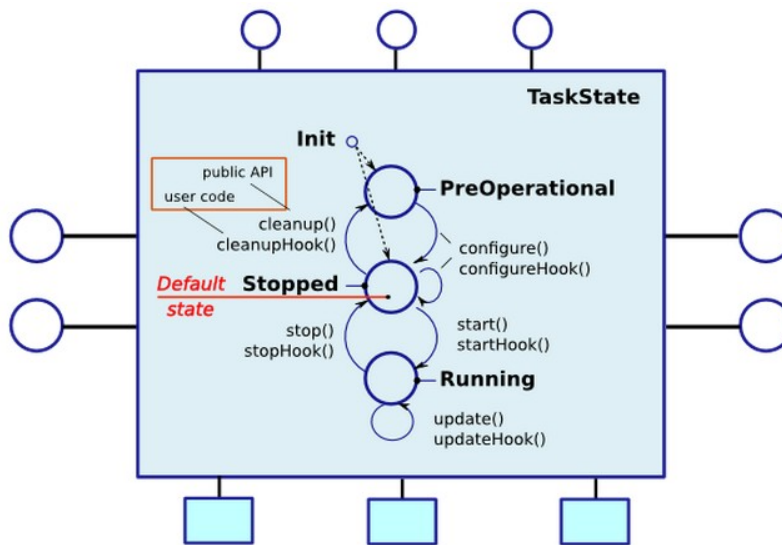
TaskBrowser је основа у структури *OROCOS* апликације преко које се остварују везе са осталим компонентама. *TaskBrowser* функционише као конзолна текст апликација, посредством које се, у виду наредби са командне линије, захтевају дефинисане функционалности компонената.

DeploymentComponent има функцију активирања и конфигурисања компонената сагласно опису садржаном у одговарајућој *XML* датотеци. Конфигурисање и реконфигурисање управљачког система подразумева остваривање веза између компонената (процеса) као и раскидање и успостављање нових посредством елемената унифицираног апликативног интерфејса (токова података, метода, команди...). Када је управљачки систем конфигурисан, својства и активности компонената се могу мењати посредством *XML* датотека.

Приликом покретања *TaskContext*-а своје извршавање роџиње и *Execution Engine*, посредством кога се прихватају команде и догађаји у току извршавања апликације. *Execution Engine* периодично проверава да ли постоје нове команде на чекању и извршава тренутно активне програме. *TaskContext* дијаграм стања приказан је на слици 4.33.

По креирању, компонента се налази у иницијалном стању (*Init* - слика 4.33). Након тога компонента може да пређе у једно од два стања: *PreOperational* или *Stopped*. Ако се прелази у *PreOperational* стање, поставља се предефинисана конфигурација компоненте позивом функције чланице *configure()*, после чега се компонента стартује позивом функције чланице *start()*. Из стања *Stopped* компонента може да пређе у стање *PreOperational* позивом функције чланице *cleanup()* или у стање *Running* позивом функције чланице *update()*. Позивом функције чланице *update()*, која је место где се имплементира извршно „језгро“ компоненте, остаје се у стању *Running*. У

стање *Stop* се прелази позивом функције чланице *stop()*, чиме се зауставља пређашња активност компоненте.



Слика 4.33 Дијаграм стања *TaskContext-a*

Са слике 4.33 се може видети да је за сваку *API (Application Programming Interface)* функцију чланицу доступна и одговарајућа корисничка функција чланица *Hook()*. Апликативни кориснички код се имплементира у наслеђеној *TaskContext* функцији чланице *Hook()*. Позивом следећих *TaskContext* функција чланица мења се стање компоненте:

- *configureHook*;
- *startHook*;
- *updateHook*;
- *stopHook*;
- *cleanupHook*.

Функционалност управљачког система за рад у реалном времену не може бити остварена без одговарајућег поретка при извршавању [89]. Узимајући то у обзир, извршавање функционалних задатака у реалном времену мора бити дефинисано коначном машином стања (*FSM - Finite State Machine*). Једна од основних предности *FSM-a* је загарантованост извршавања једног и само једног функционалног задатка у датом тренутку. Коначна машина стања представља математички модел који се користи за имплементацију, како рачунарских програма, тако и кола секвенцијалне логике. Замишљена је као апстракција машине која може у датом тренутку бити у само једном од

коначног броја стања. Стање у којем се машина стања тренутно налази се назива тренутно стање (енг. *current state*). Прелазак из једног стања у друго (енг. *transition*) може бити инициран одређеним догађајем или условом.

OROCOS садржи модул *StateMachine* који представља машину стања. *StateMachine* обезбеђује колекцију стања и функција које се активирају приликом прелазака:

- *EntryProgram* функција се позива при сваком преласку у стање за које је везана;
- *RunProgram* функција ће се позвати у тренутку када стање постане активно;
- *HandleProgram*, се извршава након што *RunProgram* извршен, али у случају да ниједан услов преласка у друга стања није испуњен;
- *ExitProgram* функције се извршава приликом напуштања стања за које је везана.

StateMachine се извршава у контексту *Execution Engine* таска, проверава услове преласка и евентуално прелази из једног у друго стање. Само је један прелаз могућ у току једног *Execution Engine* корака. И ако је број стања неограничен, свака *OROCOS* машина стања мора имати тачно једно иницијално стање (енг. *initial state*) у које ће ући приликом прве активације машине стања, као и тачно једно коначно стање (енг. *final state*) у које аутоматски прелази када се *StateMachine* заустави.

Постоје два режима рада *OROCOS* машине стања, при чему *StateMachine* може прелазити из једног мода у други у току извршавања *OROCOS* апликације:

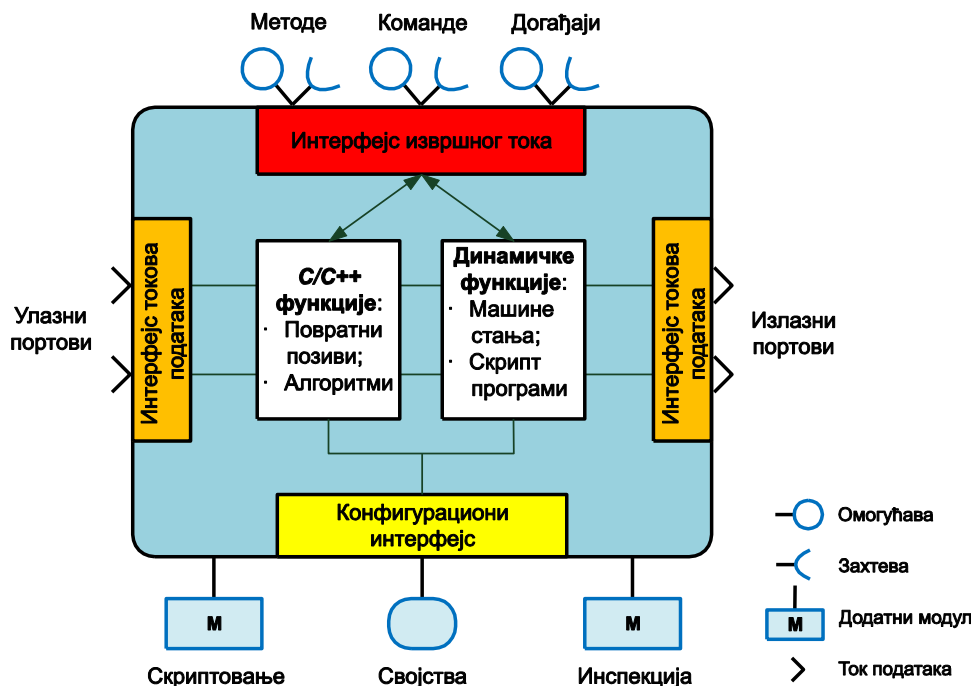
- Реактивни режим (енг. *reactive, event or trigger mode*), у који *StateMachine* улази када се активира. Након тога, промена стања може настати у два случаја:
 1. ако је иницирана неким догађајем,
 2. ако је захтевана;
- Аутоматски режим (енг. *automatic mode*), у који се улази када се, након активирања, *StateMachine* покрене. У овом моду *StateMachine* аутоматски прелази у наредна стања.

4.2.3 Имплементација софтверских компонената реконфигурабилног управљачког система применом *OROCOS*-а

Изградња реконфигурабилног управљачког система на бази *OROCOS*-а подразумева креирање софтверских компонената различите хијерархијске припадности у управљачком систему, слика 4.27. Појединачни функционални блокови се имплементирају у контексту компоненте уз дефинисање апликативног интерфејса, слика 4.34. То подразумева имплементацију управљачких алгоритама у оквиру извршног система *OROCOS* компонената уз реализацију сервисних функција и токова података апликативних интерфејса.

Свака конфигурација за управљање роботском ћелијом за обраду у реалном времену мора да садржи следећи скуп компонената:

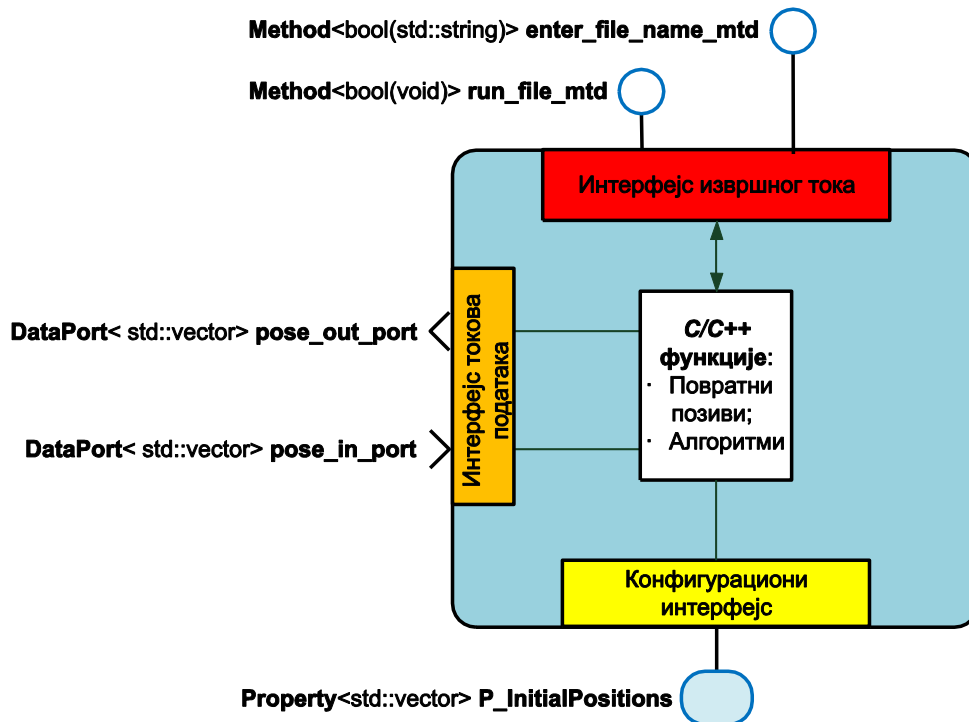
- Интерпретер међукода (*P*-кода) са одабраним модулима интерполатора;
- Кинематички модул;
- Серво контролер;
- Контролер хардвера;
- Компоненту за ручно управљање;
- Компоненту за иницијализацију оса робота.



Слика 4.34 Апликативни интерфејс *OROCOS* компоненте

На слици 4.35 је приказан пример имплементације интерпретера међукода са интегрисаним модулима интерполатора путање, *generatorPos*, као *OROCOS* компоненте за рад у реалном времену. Приказана компонента покреће активности интерполације програмиране путање позивом две методе које се извршавају у контексту компоненте која их позива:

- *Method<bool(std::string)> enter_file_name_mtd* се позива са именом програмске *P*-кôд датотеке као аргументом функције, чиме се означава који ће се програм извршавати приликом старта;
- *Method<bool(void)> run_file_mtd* функцијом се сигнализира старт матичној компоненти, интерполатору, након чега се периодично генерише нова позиција робота, сагласно програмираној путањи алата.



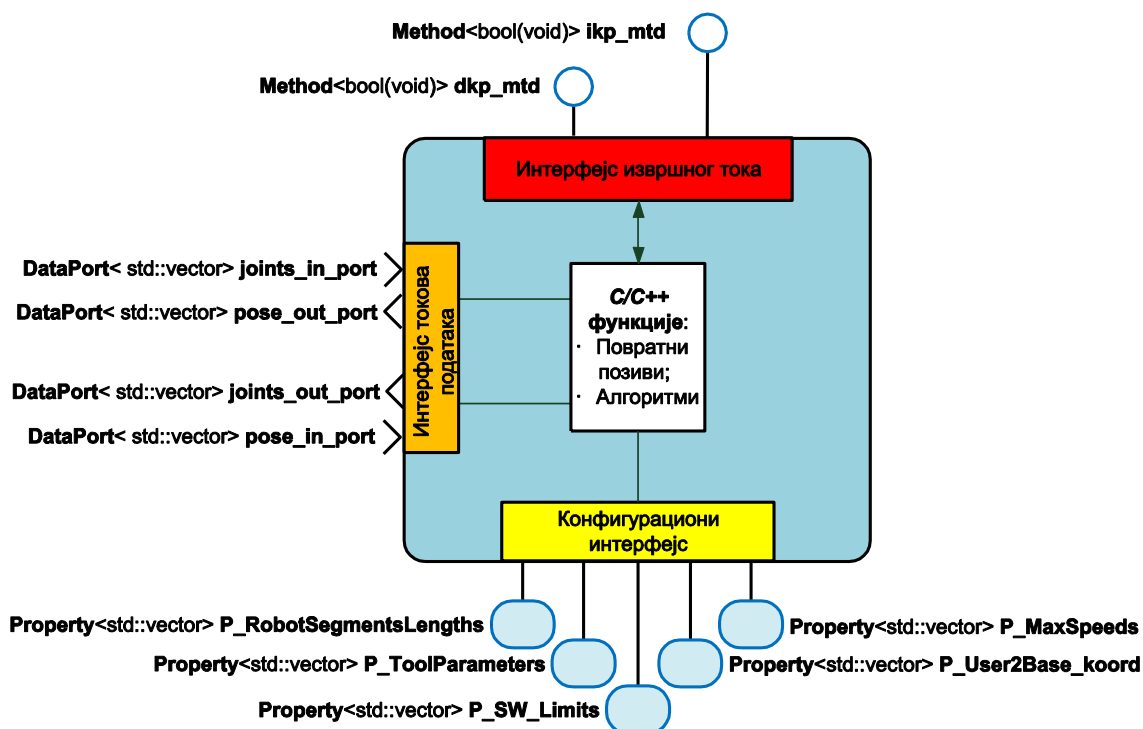
Слика 4.35 Компонента *generatorPos* - Интерполатор са интерпретером *P* кода

Интерполирана путања се у виду низа узастопних спољашњих координата (X, Y, Z, A, B, C, \dots), секвенцијално са периодом интерполације, појављује у облику тока података на *DataPort pose_out_port*. Информација о тренутној позицији алата у спољашњим координатама робота се интерполаторској компоненти достављају на *DataPort pose_in_port*,

посредством хардверски оријентисаних компонената система и кинематичког модула.

По иницијализацији роботске ћелије за обраду, иницијалне вредности спољашњих и унутрашњих координата се посредством конфигурационог интерфејса и својства *P_InitialPositions* преносе компоненти *generatorPos*. Иницијалне позиције оса су од значаја приликом покретања апликативног програма и узимања почетне позиције у тренутку отпочињања интерполације.

Слика 4.36 приказује компоненту *kinematics* која садржи алгоритме за решавање инверзне и директне кинематике роботске ћелије за обраду.



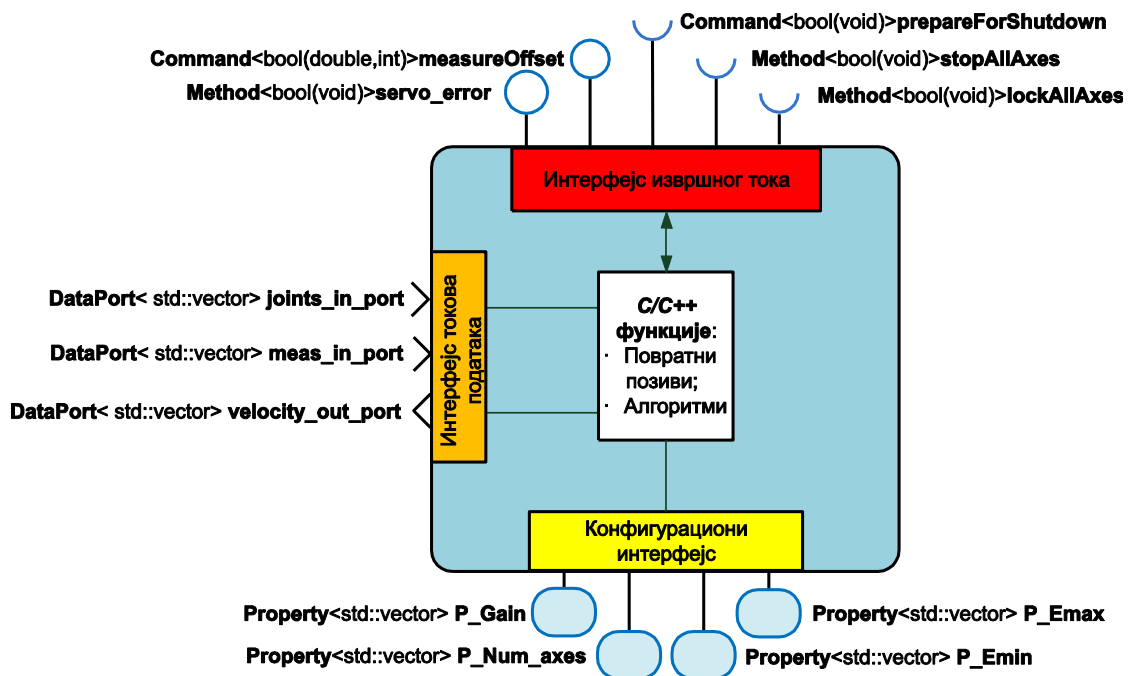
Слика 4.36 Компонента *kinematics* - Имплементација функција инверзне и директне кинематике

Позивом метода *ikp_mtd* или *dkp_mtd* компоненти *kinematics* се сигнализира почетак активности. Сигнализацију обавља компонента која је посредством токова података доставила неопходне податке за обављање кинематичких функција - спољашње координате на *DataPort pose_in_port* ако је захтеван прорачун инверзне кинематике или унутрашње координате на *DataPort joint_in_port* уколико је захтеван прорачун директне кинематике. С обзиром на то да компонента *kinematics* ради у режиму *SlaveActivity*, покреће

је позив функције *Update()* од стране компоненте која је претходно упутила захтев за извршење кинематичке функције. У функцији *updateHook()*, зависно од постављеног маркера, позива се одговарајућа кинематичка функција. Како би се осигурао ексклузиван приступ функцијама у реалном времену, критични делови кода се штите *mutex*-има. Резултат се у виду тока података поставља на *joint_out_port* тј. *pose_out_port* за функцију инверзне, односно директне кинематике, респективно.

С обзиром на то да се сви кинематички параметри чувају у *XML* датотеци, омогућено је да се исти могу мењати без поновне компилације, чиме је обезбеђена конфигурабилност кинематичке компоненте.

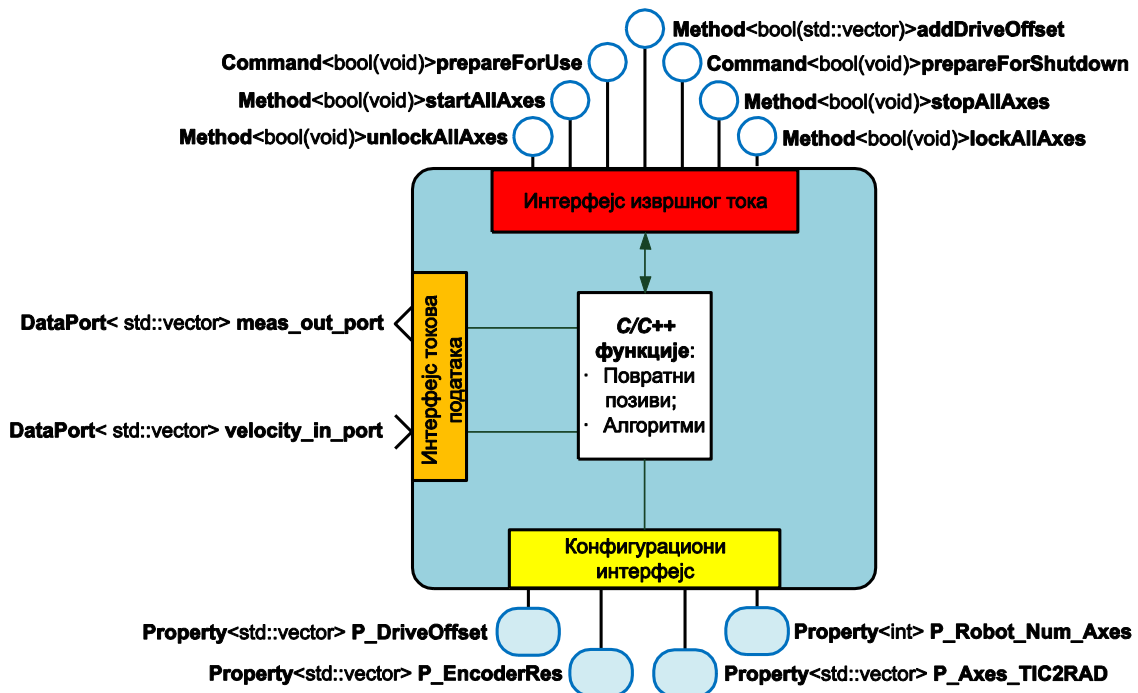
На слици 4.37 је приказана компонента *nAxesContorllerPos* која обавља функцију серво контролера позиције. Извршава се као периодична компонента са највишом учестаношћу у систему. Врши поређење задате позиције оса из интерполатора (*DataPort joints_in_port*) са очитаним вредностима са мерних система (*DataPort meas_in_port*) и на основу серво алгоритма формира задату вредност брзина оса (*DataPort velocity_out_port*). Компонента *nAxesContorllerPos* садржи и алгоритам за фину интерполацију, који генерише додатне сегменте путање између интерполираних тачака.



Слика 4.37 Компонента *nAxesContorllerPos* - Имплементација серво контроле позиција оса

За формирање серво алгоритма од значаја је својство P_Gain које између осталог може садржати вредности појачања, интегралног и/или диференцијалног дејства позиционе серво петље. Број управљаних оса садржан је у својству P_Num_axes . У случају да разлика задате и измерене вредности позиције осе изађе из опсега вредности апсолутне разлике $|P_Emax - P_Emin|$, позивају се методе *stopAllAxes*, *lockAllaxes* и команда *prepareForShutdown* секвенцијално у циљу заустављања рада система. Остале компоненте у систему имају могућност да испитају да ли је наступила серво грешка позивом методе *servo_error*. Команда *measureOffset* се позива приликом мерења офсета брзине у систему, у току извршавања иницијалних алгоритама.

На слици 4.38 је приказана компонента *robotHardware* која представља апликативни интерфејс ка картицама за улазно/излазне операције и актуаторима оса. Компонента *robotHardware* енкапсулира имплементацију драјвера спрежних картица за улазно/излазне операције. На тај начин обезбеђује јединствен апликативни интерфејс ка осталим компонентама система без обзира на коришћени спрежни хардвер. Оваква имплементација омогућава интеграторима система да без интервенција на језгру

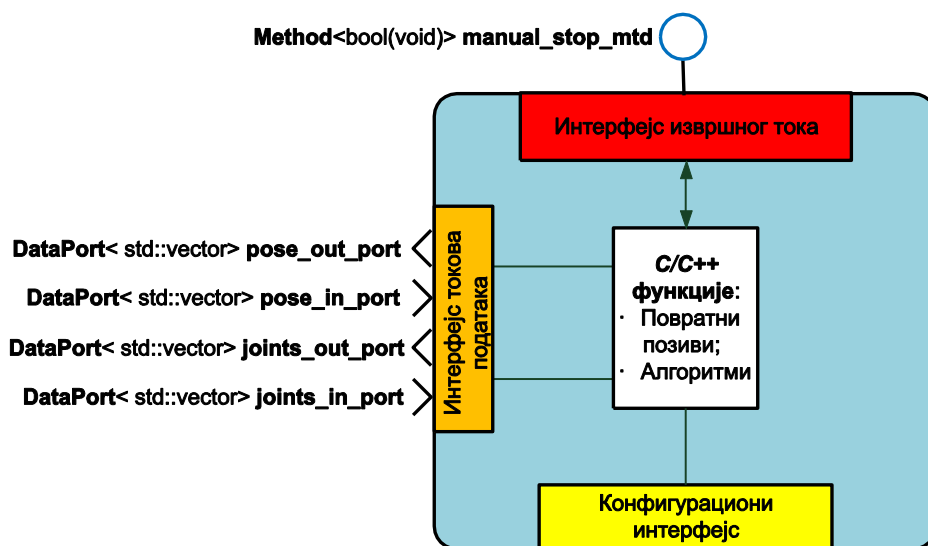


Слика 4.38 Компонента *robotHardware* - Апликативни интерфејс ка спрежним картицама и актуаторима

управљачког софтвера, конфигуришу различите хардверске интерфејсе за везу са погонима роботске ћелије. Посредством тока података *velocity_in_port* хардверској компоненти се достављају информације о жељеној брзини оса, које иста прослеђује одговарајућем драјверу. Информације о тренутним позицијама оса се посредством тока података *meas_out_port* прослеђују осталим компонентама система.

Придružене команде и методе, које се могу видети на слици 4.38, представљају механизам секвенцијалне контроле оса роботске ћелије, као и евентуално покретање безбедносних процедура. За иницијално конфигурисање компоненте *robotHardware*, задужена су придružена својства. Својство *P_DriveOffset* омогућава анулирање напонског офсета, који се може појавити на излазима *D/A* конвертора у случају када се брзина регулаторима мотора задаје у виду пропорционалног напона. Својство *P_EncoderRes* поставља резолуцију мерних система свих оса роботске ћелије. Посредством својства *P_Axes_TIC2RAD* постављају се константе преноса између мерних система и оса. Својство *P_Robot_Num_Axes* даје информацију систему о броју активних оса роботске ћелије.

Слика 4.39 представља компоненту за ручно управљање *manualMode*. Осама роботске ћелије се може управљати у ручном режиму задавањем унутрашњих и/или спољашњих координата у зависности од конфигурације. У случају да се управља унутрашњим координатама, задате вредности

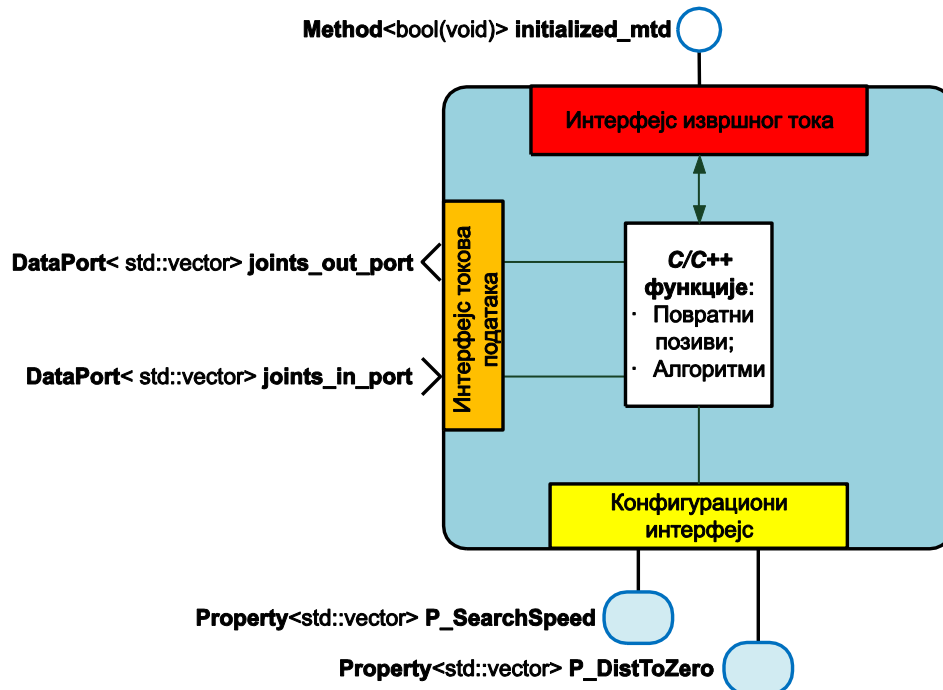


Слика 4.39 Компонента *manualMode* - омогућава ручно управљање осама роботске ћелије

позиција се емитују посредством тока података *joints_out_port*. Када се управља спољашњим координатама, исте се посредством тока података *pose_out_port* најпре морају проследити компоненти *kinematics*, а затим у облику вредности унутрашњих координата предати другом делу система (серво компоненти). По аналогији са компонентом интерполатора, нове вредности позиција се генеришу са периодом интерполационог такта.

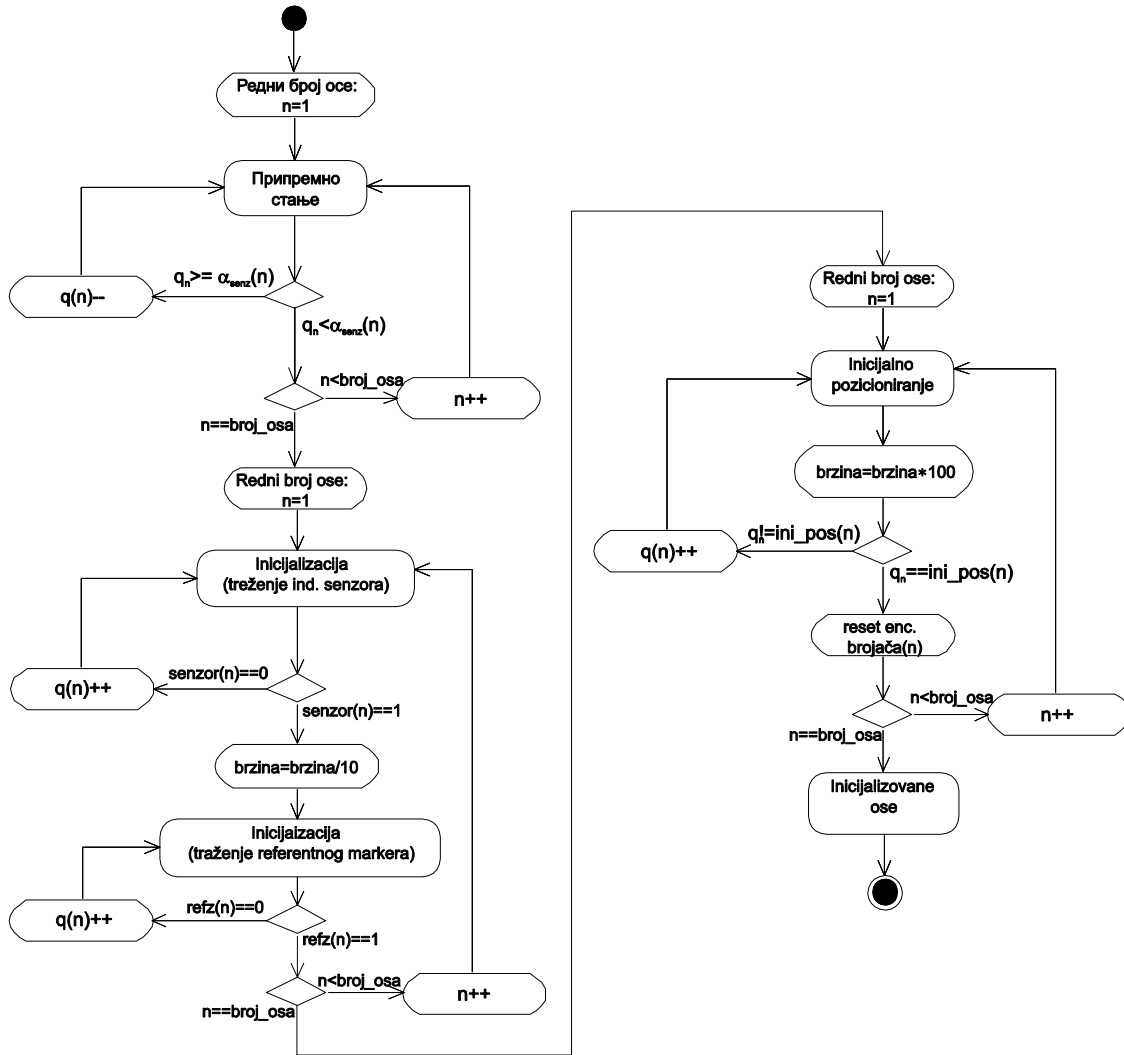
Са стране крајњег корисника, на располагању су разне врсте интерфејса за контролу ручног вођења оса роботске ћелије. Могу се користити стандардне периферије *PC* рачунара, као што су тастатура, палица за игру (енг. *joystick*) и слично, али се могу повезати и специјализовани хардверски интерфејси, попут инкременталног точића (енг. *hand wheel*) и/или ручне палице. Прекид рада компоненте *manualMode* се захтева позивом методе *manual_stop_mtd*, после чега се раскидају претходно успостављене везе са компонентама система.

На слици 4.40 је приказана компонента *initRobot*, која се користи током одређивања јединствених позиција оса роботске ћелије и иницијалног позиционирања.



Слика 4.40 Компонента *initRobot* - омогућава иницијализацију оса роботске ћелије

Слика 4.41 представља дијаграм активности, према коме се иницијализација роботске ћелије се обавља у три фазе. Првој фази претходи позиционирање оса роботске ћелије у зону осетљивости индуктивних сензора, што представља недвосмислену позицију за отпочињање иницијализације.



Слика 4.41 Дијаграм активности функције *updateHook* садржане у компоненти *initRobot*

По старту иницијализације компонента *initRobot* улази у припремно стање. У овом стању систем остаје док све осе, крећући се у негативном смеру, не изађу из зоне осетљивости индуктивних сензора. Након тога се улази у стање иницијализације (тражење референтног сензора). Сада се мења смер кретања и осе се на исти начин крећу док не уђу у зону осетљивости сензора, када се прелази у следеће стање иницијализације (тражење референтног маркера). Покрећући осе у истом смеру, функција

`updateHook()` покушава да детектује сигнал референтног маркера оптичког мерног система.

Присуство референтног маркера у зони осетљивости индуктивног сензора, у позитивном смеру кретања, дефинише јединствену позицију осе у простору кретања. По детекцији јединствених позиција свих оса роботске ћелије, прелази се у наредно стање, иницијално позиционирање, које има за задатак да осе роботске ћелије одведе у нулте референтне позиције и ресетује енкодерске бројаче.

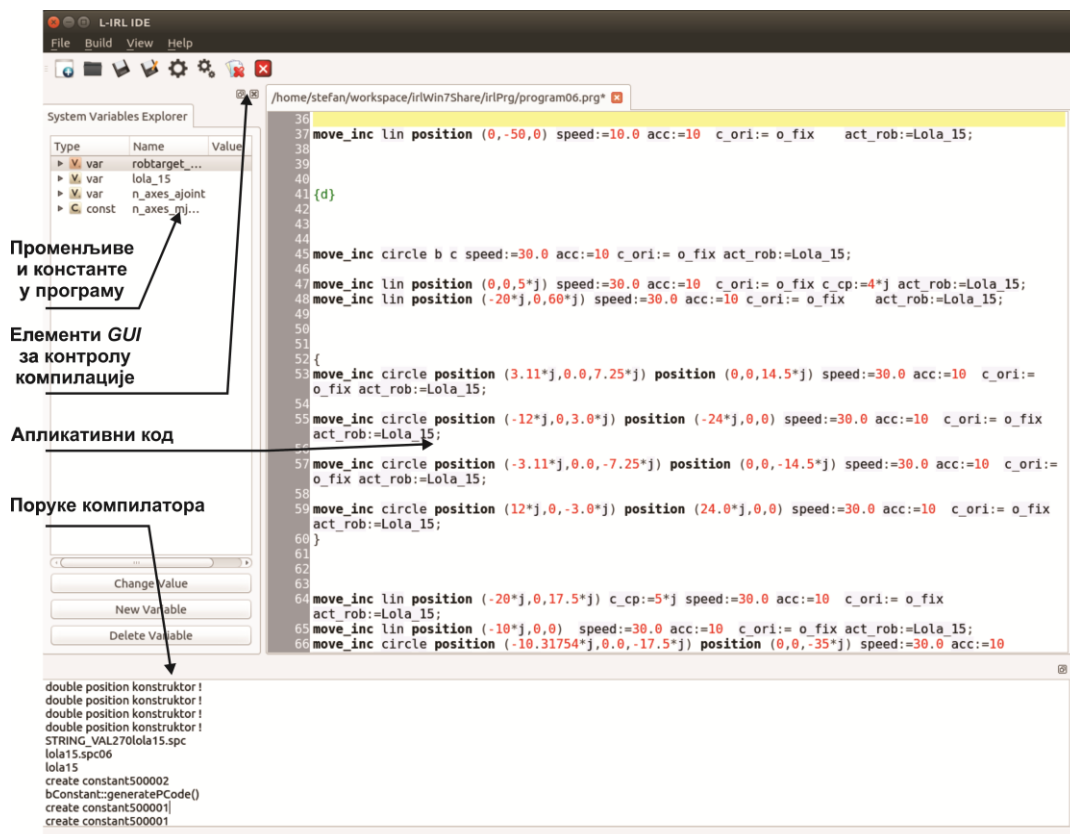
4.2.4 *Off-line* подсистем за програмирање и симулацију обраде

Off-line систем за програмирање и симулацију рада роботских ћелија за обраду обједињује неколико функционалних блокова. Основу система чине већ описани управљачки алгоритми са компилатором програмског кода, прилагођени за извршавање на десктоп оперативном систему, слика 4.27. Како би *off-line* подсистем могао да комуницира са спољашњим светом, корисником са једне стране и *real-time* управљачким подсистемом са друге стране, језгру система су придружени софтверски интерфејси и одговарајући комуникациони протоколи који су развијени коришћењем савремених софтверских технологија отворене архитектуре.

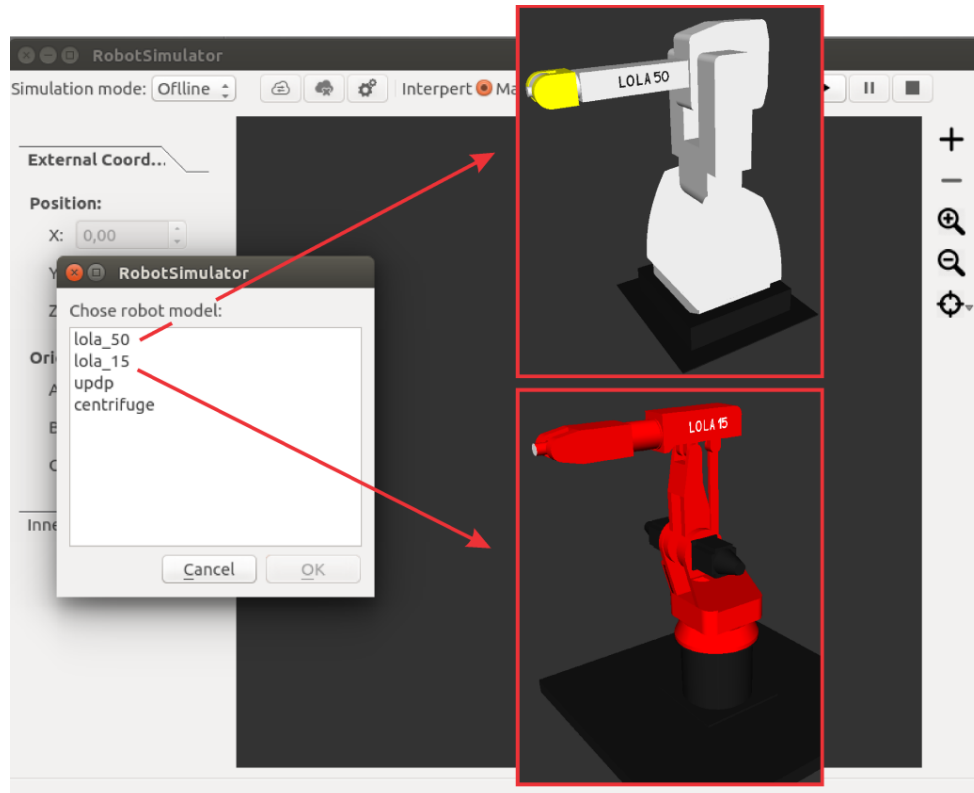
Са стране корисника апликације налази се графички кориснички интерфејс (*GUI*) који садржи:

- Едитор програмског кода са визуелним елементима за контролу компилације програмског кода, слика 4.42;
- Интерфејс за избор конфигурације роботске ћелије, слика 4.43;
- *3D* виртуелну роботску ћелију за *off-line* симулацију и праћење рада робота у реалном времену, слика 4.44.

За реализацију корисничког дела интерфејса коришћени су алати *Qt* и *OpenGL*. *Qt* библиотеке су коришћене за реализацију едитора кода, дијалога, контролних елемената, прозора за *3D* приказа виртуелног робота и тренутних позиција оса робота. *OpenGL* библиотеке се користе за моделирање и анимацију виртуелних робота и виртуелних роботских ћелија. Конфигурисање виртуелних роботских ћелија своди на програмирање

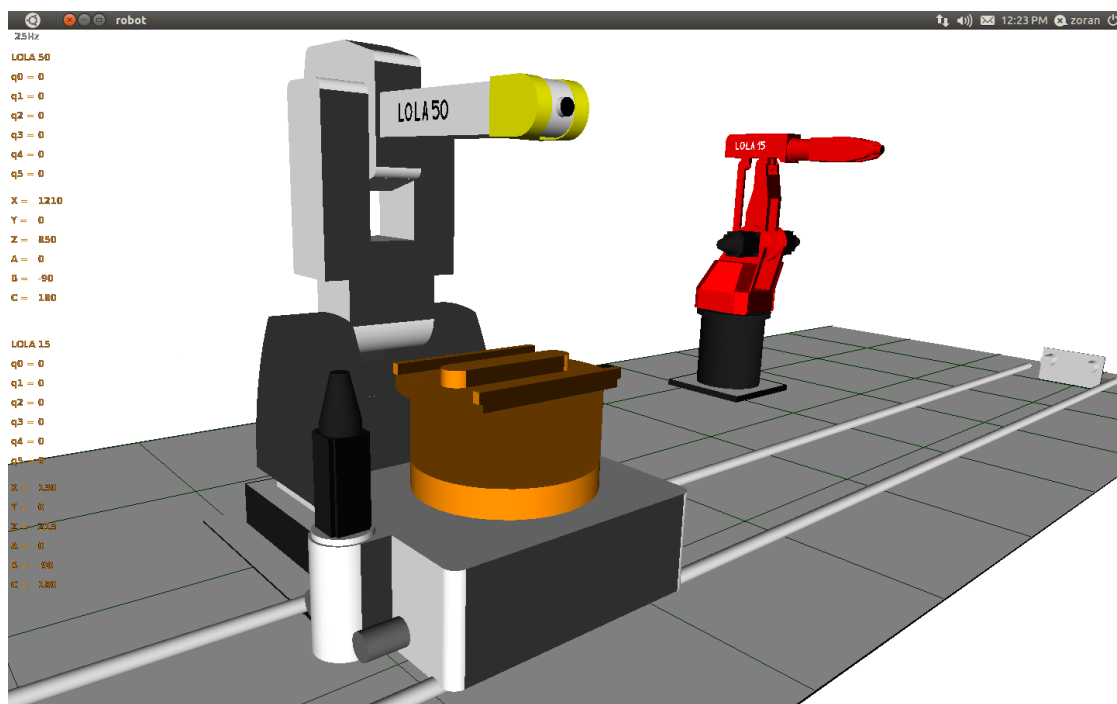


Слика 4.42 Едитор програмског кода off-line система



Слика 4.43 Конфигурациони интерфејс

координата геометријских примитива које се групишу у склопове виртуелног робота, еквивалентно поступку који се користи код софтвера *EMC2*, слика 4.7.



Слика 4.44 Виртуелна ћелија са два робота

Пример дела програма за једну осу, моделирану у *OpenGL*-у применом елементарних 3D геометријских примитива, има следећи облик:

```
CylinderX(-80, 95, -40, 95);
CylinderX(40, 95, 80, 95);
Box(-80, -95, -195, -40, 95, 0.0);
Box(40, -95, -195, 80, 95, 0.0);
Box(-40, -95, -195, 40, 95, -95);
CylinderZ(-220,85,-195,85);
```

Као што се може приметити, за реализацију виртуелних робота се користе предефинисане класе *C++* програмског језика, које се инстанцирају позивом одговарајућег конструктора са параметрима геометријског примитива који се жели добити.

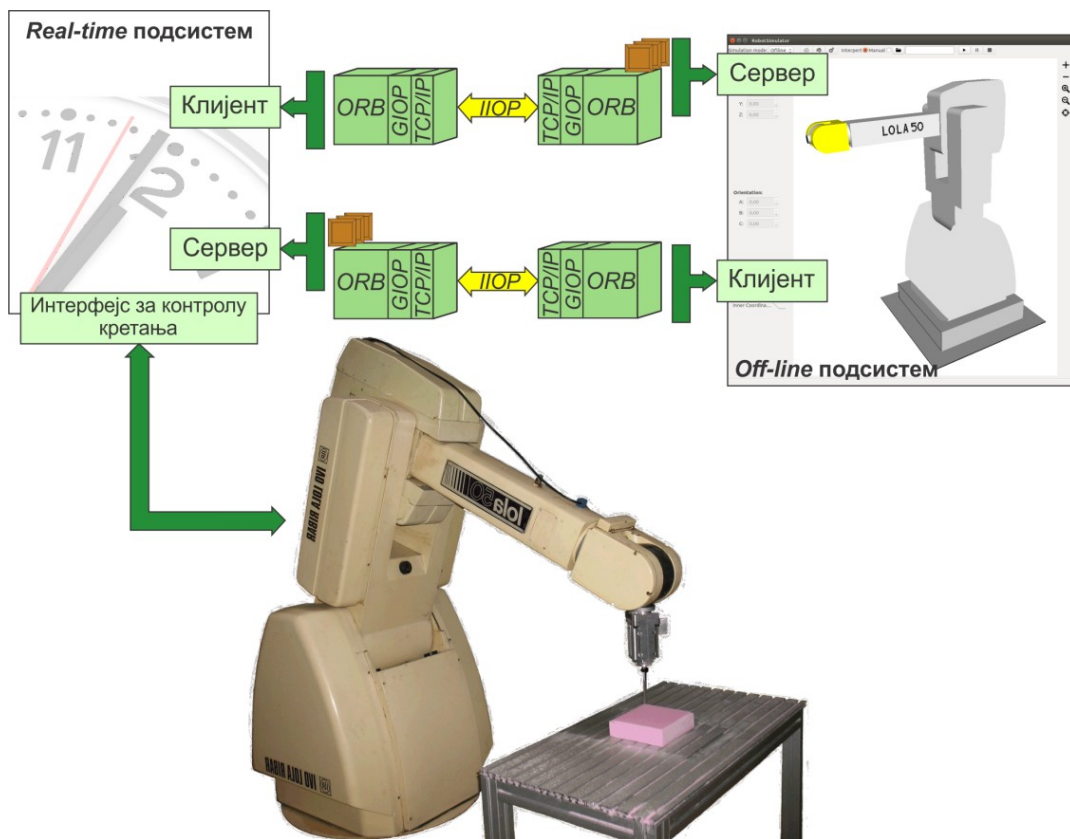
Софтверски управљачки подсистеми, *real-time* и *off-line*, могу бити базирани на једном рачунару или на два умрежена рачунара. У оба случаја се комуникација између подсистема остварује посредством клијент-сервер софтверских модула у циљу обављања следећих задатака:

- Пренос верификованог програма обраде, у виду међукода (*P*-кода), са *off-line* подсистема на *real-time* управљачки подсистем;

РАЗВОЈ МЕТОДА ПРОЈЕКТОВАЊА УПРАВЉАЧКОГ СИСТЕМА...

- Пренос статусних команди које корисник, посредством *off-line* подсистема, упућује *real-time* управљачком подсистему, односно роботској ћелији;
- Пријем статусних информација са роботске ћелије на страни *off-line* подсистема;
- Пријем информација о позицијама оса у реалном времену на страни *off-line* подсистема, ради праћења рада робота посредством 3D виртуелне роботске ћелије.

Комуникација клијент-сервер између *off-line* и *real-time* подсистема се обавља посредством *CORBA* протокола, слика 4.45. При томе је *ORB* (**O**bject **R**equest **B**roker) централни део протокола, посредством кога се обавља комуникацију између објеката. *ORB* управља трансформацијом структура података, у секвенцу бајтова и обрнуто, који се преносе посредством комуникационог канала (мреже). *ORB* се понаша као магистрала преко које објекти међусобно комуницирају, локално или са удаљених хардверских ресурса. Комуникација између *ORB*-ова се обавља посредством *GIOP*-а



Слика 4.45 Комуникација између *real-time* и *off-line* подсистема

(*General Inter-ORB Protocol*). По *CORBA* спецификацији сви *ORB*-ови морају имати имплементиран *GIOP*, или његову имплементацију преко *TCP/IP*-а (*Transmission Control Protocol/Internet Protocol*), која је означена са *IIOIP* (*Internet Inter-ORB Protocol*).

Предложени управљачки систем се базира на двосмерној, дуплекс комуникација између два подсистема, *off-line* и *real-time*, слика 4.45. Када се обавља пренос датотека, које садрже програме обраде у виду међукóда, користи се комуникациони канал у коме се *off-line* подсистем се понаша као клијент, док је *real-time* подсистем сервер. Исти комуникациони канал се користи за пренос команди које покрећу одређене акције у управљачком систему, односно роботској ћелији. У измењеним улогама, посредством другог комуникационог канала, *off-line* подсистем прима статусне сигнале и податке о позицијама оса робота.

На основу добијених података од *real-time* подсистема о позицијама оса, у *off-line* систему се врши покретање оса виртуелне роботске ћелије. Акција се покреће позивом функције *setAxis()* чији оперативни део, за случај робота са 6 степени слободе, има следећи облик:

```
frame(1)-> setRotation(Quaternion(Vec(0.0, 1.0, 0.0), Axis1));
frame(2)-> setRotation(Quaternion(Vec(1.0, 0.0, 0.0), Axis2));
frame(3)-> setRotation(Quaternion(Vec(1.0, 0.0, 0.0), Axis3));
frame(4)-> setRotation(Quaternion(Vec(0.0, 0.0, 1.0), Axis4));
frame(5)-> setRotation(Quaternion(Vec(1.0, 0.0, 0.0), Axis5));
frame(6)-> setRotation(Quaternion(Vec(0.0, 0.0, 1.0), Axis6));
```

...

Низ *frame* представља низ објеката класе *ManipulatedFrame* у којој су дефинисани алгоритми за покретање примитива, ротације и транслације, за задати износ. Функција чланица класе *ManipulatedFrame*, *setRotation* (*const Quaternion &rotation*) поставља вредност ротације задату аргументом *Quaternion*. Аргумент *Quaternion* дефинише осу и угао ротације.

4.2.5 Приказ морфолошког метода за конфигурисање и реконфигурисање управљачког система на бази *OROCOS*-а

Процедура за конфигурисање, односно реконфигурисање предложеног управљачког система се обавља у следећим фазама:

- Иницијално конфигурисање;

- Статичко реконфигурисање
- Динамичко реконфигурисање;

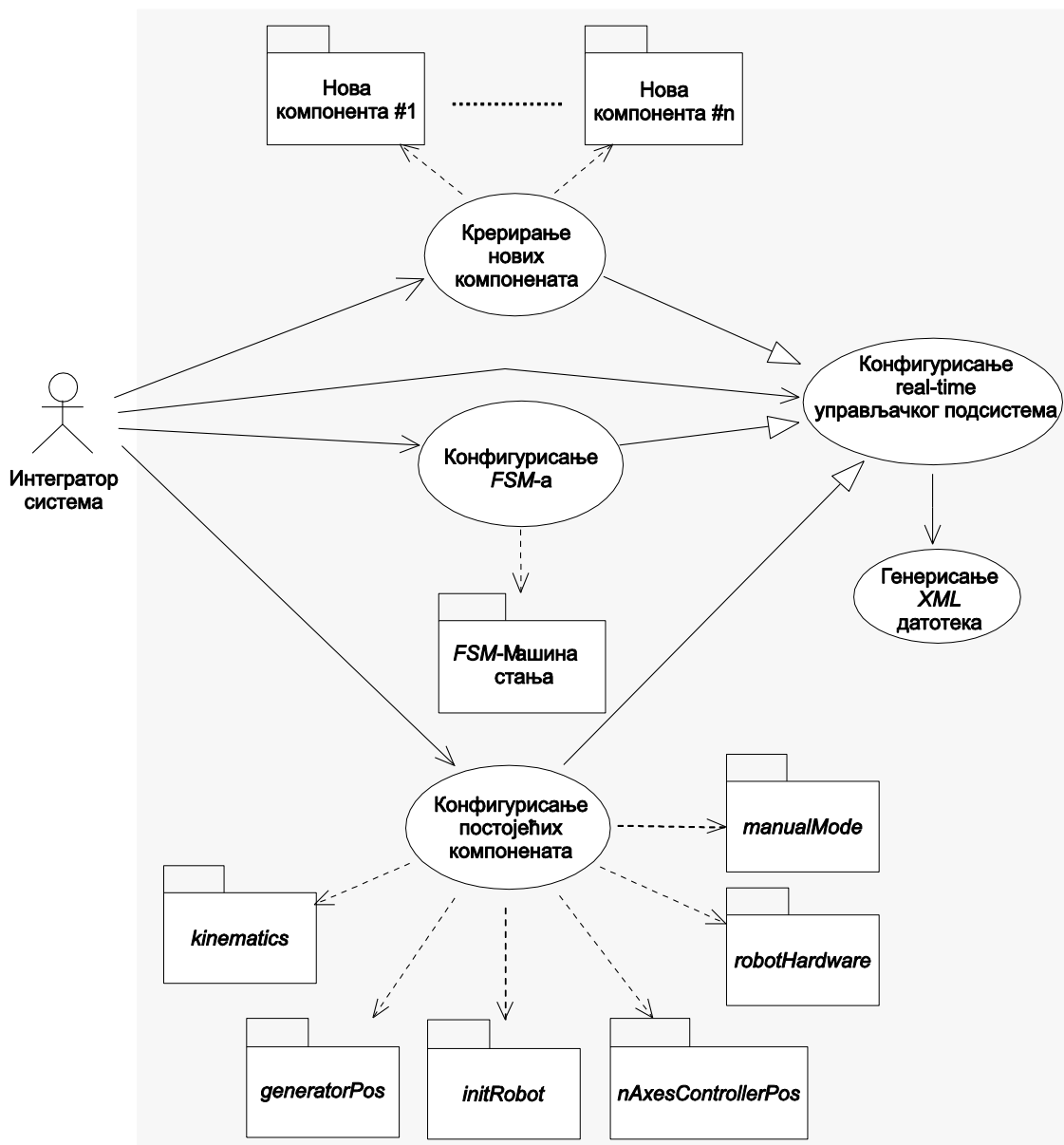
У фази иницијалног конфигурирања је неопходно користити развојне компилаторске алате, ради изградње компонената карактеристичних за одабран скуп конфигурација роботске ћелије. Фазе статичког и динамичког реконфигурисања се спроводе избором и/или модификацијама конфигурационих датотека, за шта није потребно имати програмерска умећа.

Фаза иницијалног конфигурирања

Под појмом иницијалног конфигурирања управљачког система подразумева се формирање коначног броја иницијалних конфигурација за управљање реконфигурабилном роботском ћелијом за обраду. Разматрани управљачки систем треба да обухвати све софтверске компоненте потребне за управљање одабраним конфигурацијама роботске ћелије за обраду, као и захтеваним апликацијама обраде. Према томе, задатак интегратора система, који врши иницијално конфигурирање *real-time* управљачког подсистема, треба да обухвати:

- Израду одговарајућег броја кинематичких компонената (нпр. к3.1 до к3.5) роботске ћелије са припадајућим *XML* датотекама својстава;
- Одабир подмодула интерполатора и интеграцију истих у јединствену компоненту;
- Модификацију компоненте *initRobot* уз имплементацију *XML* датотека својстава за све дефинисане конфигурације;
- Модификацију компоненте *nAxesContorllerPos* уз имплементацију *XML* датотека својстава за све дефинисане конфигурације;
- Конфигурисање компоненте *robotHardware* према захтеваном броју активних оса;
- Конфигурисање компоненте *manualMode* према захтеваном броју активних оса;
- Интеграцију компонената *real-time* подсистема посредством *XML* конфигурационог интерфејса;
- Конфигурисање машине стања посредством *XML* интерфејса.

UML дијаграм случајева коришћења у току конфигурисања *real-time* управљачког подсистема приказан је на слици 4.46.

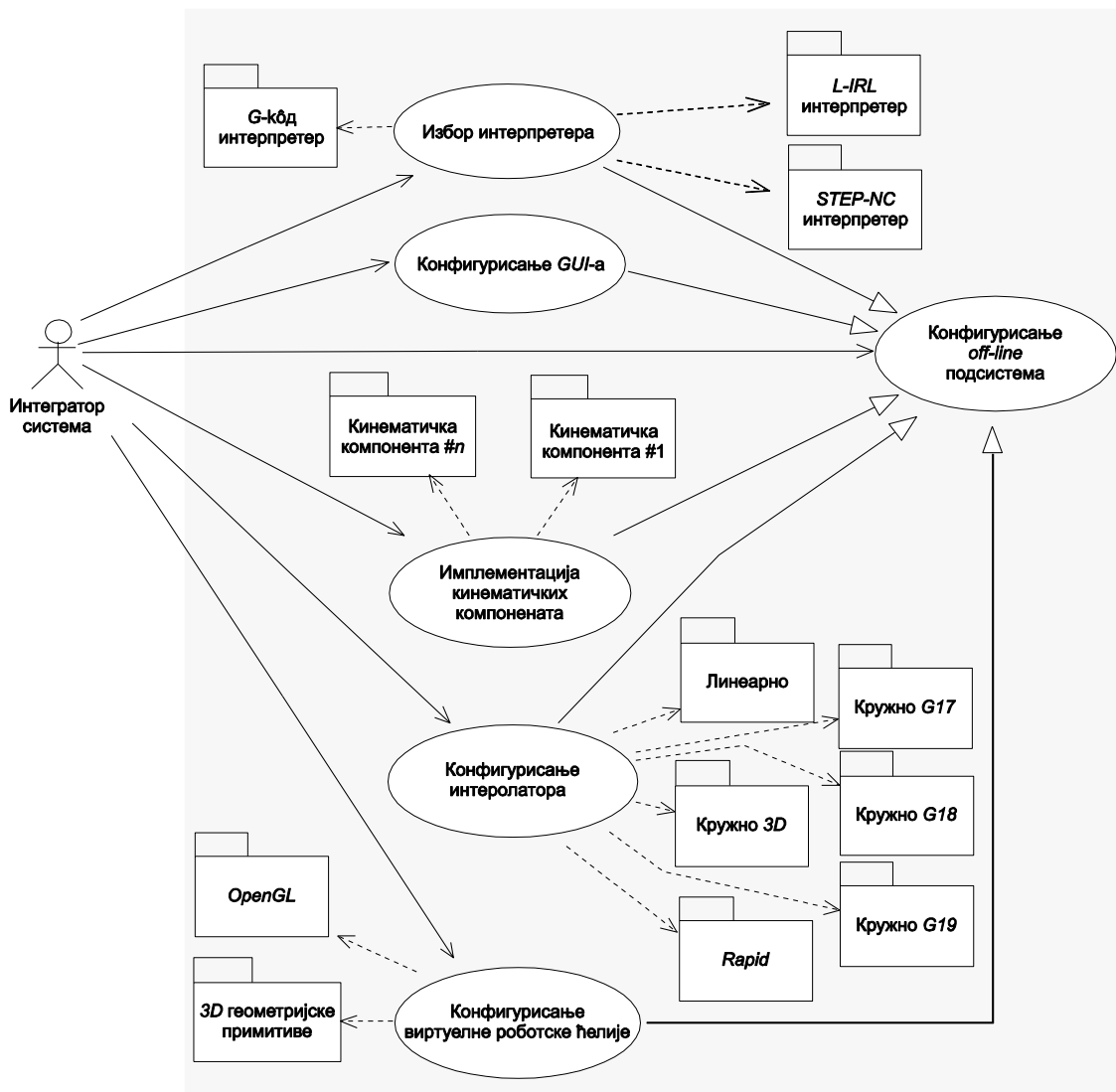


Слика 4.46 Иницијално конфигурисање *real-time* управљачког подсистема

У фази иницијалног конфигурисања формира се више конфигурација. Информације о компонентама које улазе у састав једне конфигурације, као и о начину њихове интерконекције, садржане су у придруженој конфигурационој XML датотеци.

Упоредо са конфигурисањем *real-time* подсистема, мора се припремити и *off-line* подсистем како би подржао предвиђене конфигурације роботске ћелије за обраду, слика 4.47. У том смислу неопходне су следеће активности:

- Избор интерпретера корисничког програма (програма обраде);
- Имплементација кинематичких компонената *off-line* подсистема;
- Прилагођење интерполатора захтевима корисника и конфигурацијама роботске ћелије;
- Прилагођење графичког корисничког интерфејса (*GUI*) за избор конфигурације;
- Конфигурисање виртуелних роботских ћелија за симулацију програма обраде и праћење рада ћелије у реалном времену.



Слика 4.47 Иницијално конфигурисање *off-line* подсистема

Фаза статичког реконфигурисања

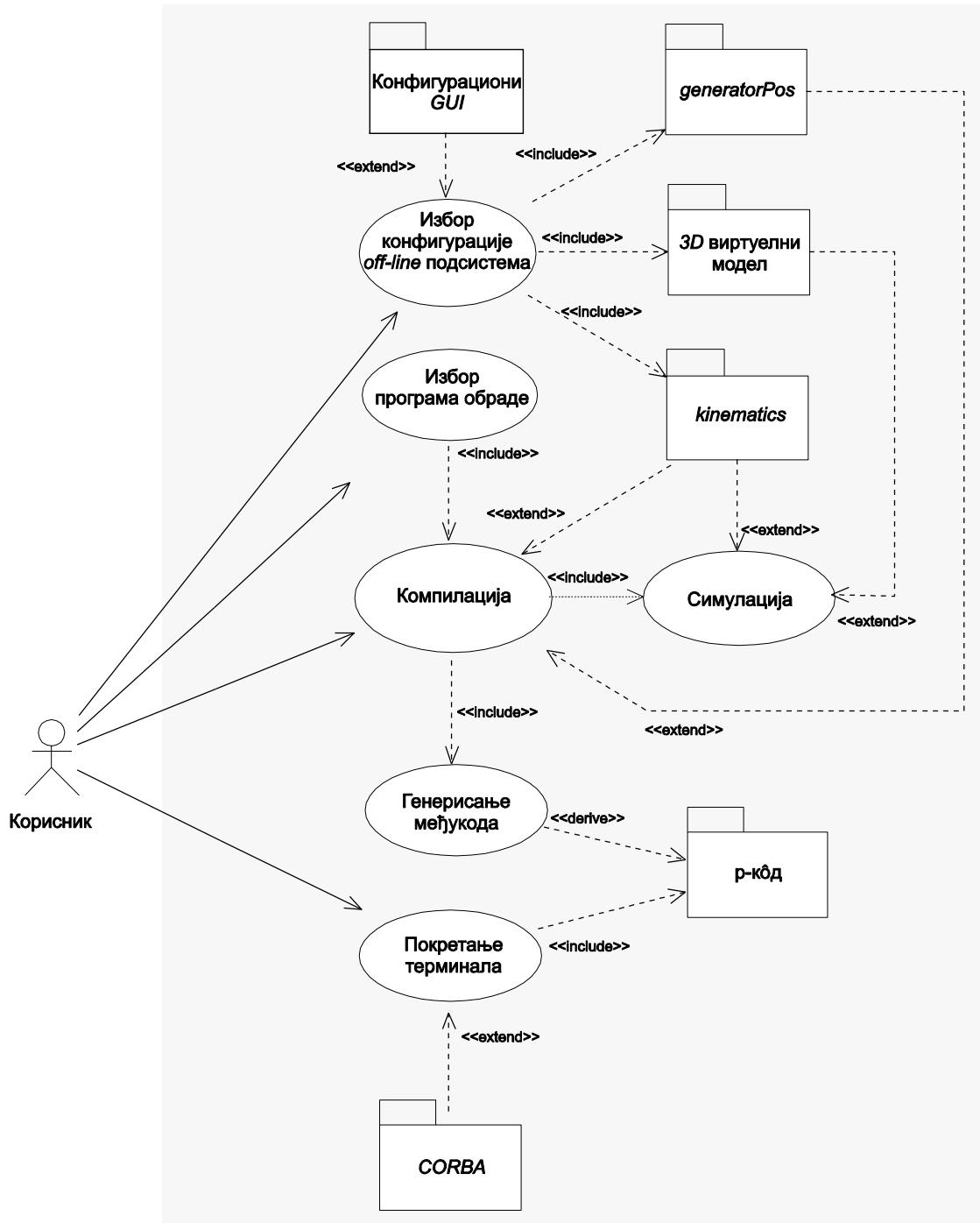
Фаза статичког реконфигурисања управљачког система подразумева избор управљачке конфигурације роботске ћелије за обраду пре него што систем постане активан. Ова фаза подразумева предузимање неколико активности у циљу добијања функционалног управљачког система. Под претпоставком да је систем иницијално конфигуриран, односно да су дефинисане могуће конфигурације роботске ћелије и управљачког система, фаза статичког реконфигурисања подразумева следеће активности у *off-line* подсистему:

- Избор конфигурације роботске ћелије у одговарајућем прозору приликом подизања *off-line* подсистема. На тај начин је селектован одговарајући кинематички модул, скуп подмодула интерполатора, као и одговарајућа виртуелна роботска ћелија за праћење симулације програма обраде;
- Избор одговарајућег програма обраде;
- Компилацију програма обраде уз интерполацију инструкција кретања и инверзну кинематичку трансформацију у циљу провере радног простора и детекције колизионих ситуација;
- Извођење визуелних симулације програма обраде на *3D* виртуелној роботској ћелији у циљу провере програма обраде и конфигурације роботске ћелије.
- Генерисање и пренос међукôда на *real-time* подсистем посредством комуникационих линија и *CORBA* протокола;
- Прелазак у режим терминала за контролу и праћење рада роботске ћелије посредством *3D* виртуелне роботске ћелије у реалном времену.

На слици 4.48 је приказан *UML* дијаграм случајева коришћења у току статичког конфигурирања *off-line* подсистема, компилације и симулације програма обраде.

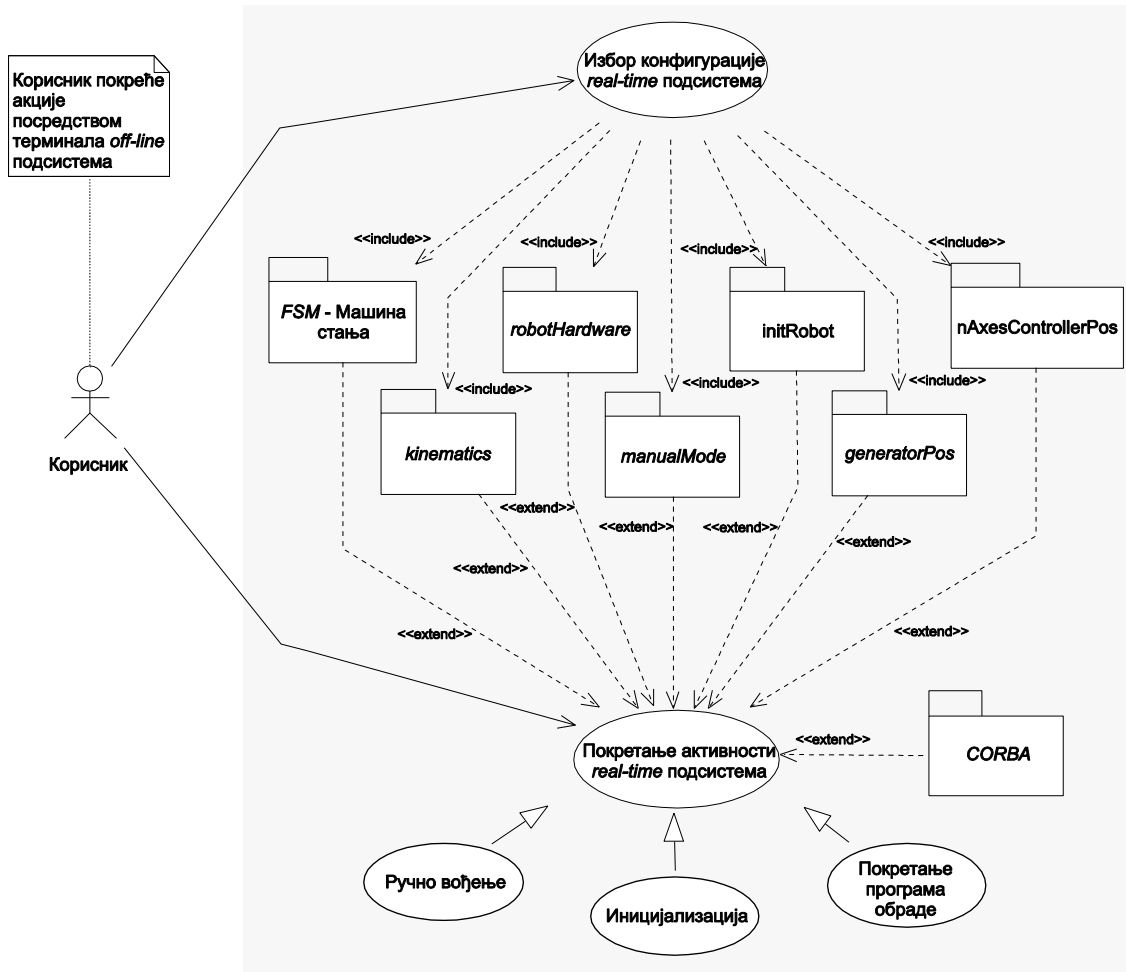
На страни *real-time* подсистема се избором одговарајуће *XML* конфигурационе датотеке, пре покретања софтвера, одређује управљачка конфигурација роботске ћелије. На тај начин се одређују компоненте са *XML*

својствима које улазе у састав управљачке конфигурације, везе између њих, као и машина стања која ће водити систем током рада.



Слика 4.48 Пример статичког конфигурисања *off-line* подсистема, компилације и симулације програма обраде

На слици 4.49 је приказан *UML* дијаграм случајева коришћења у току статичког реконфигурисања *real-time* управљачког подсистема.



Слика 4.49 Пример статичког реконфигурисање *real-time* управљачког подсистема

Фаза динамичког реконфигурисања

До сада, када је било речи о конфигурисању и статичком реконфигурисању управљачког система, подразумевано је да се у циљу промене конфигурације, одговарајуће акције спроводе над оба подсистема, *real-time* и *off-line*. Када реч о динамичкој реконфигурабилности сматраћемо да се акције у циљу промене конфигурације управљачког система спроводе у реалном времену, без искључења погона, без деалоцирања меморије рачунарског система коју су заузеле активне управљачке компоненте. Другим речима, динамичко реконфигурисање се спроводи у тренуцима када је управљачки систем активан. С обзиром на чињеницу да је потреба за променама у кинематици роботске ћелије током обраде главни разлог за динамичком изменом конфигурације управљачког система, у даљем тексту

ће бити речи о механизму за динамичко реконфигурисање кинематичке компоненте предложеног управљачког система. Структура роботске ћелије се може променити на више начина. Са становишта реконфигурисања управљачке јединице од значаја су следећа два начина:

- Промене кинематичких својстава које се рефлектују на геометријске параметре, као што су дужине сегмената и/или број степени слободе система;
- Промене у морфолошкој структури роботске ћелије.

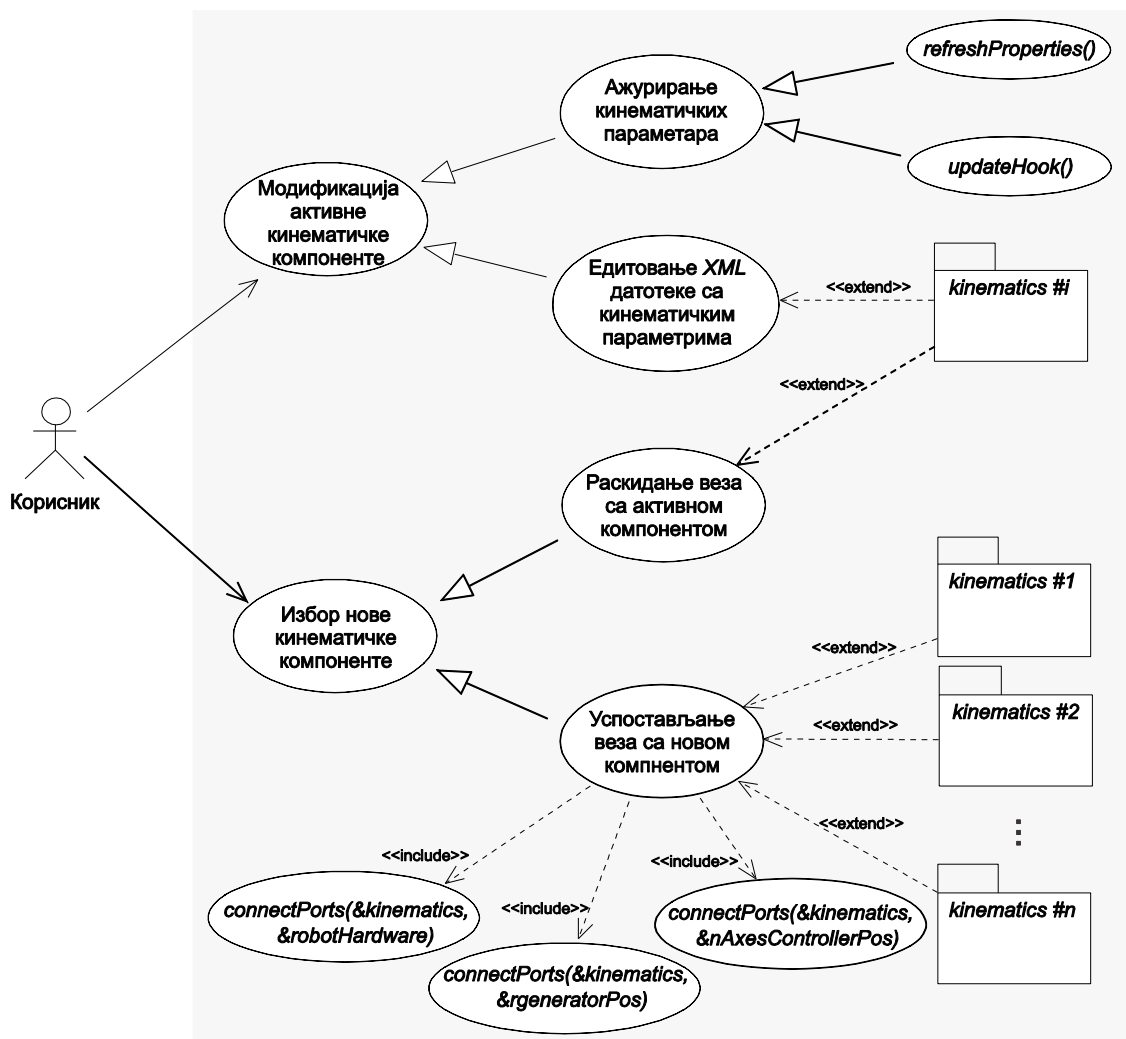
Поменути начини реконфигурисања роботске ћелије дефинишу два начина динамичке реконфигурабилности *real-time* управљачког подсистема:

- Модификацијом кинематичке компоненте;
- Заменом кинематичке компоненте.

Под модификацијом кинематичке компоненте подразумева се измена параметара кинематичких функција, садржаних у *XML* датотеци својстава, која треба да испрати промене геометријских параметара механизма роботске ћелије. С обзиром на то да *XML* датотека представља текстуалну датотеку, измена њеног садржаја се може обавити посредством неког едитора текста или специјализованог софтвера израђеног за дату намену. Како би се модификације датотеке рефлектовале на функционалност кинематичке компоненте, морају бити испраћене одговарајућим системским позивима. У најновијој верзији *OROCOS*-а, за освежавање својстава компоненте користи се системски позив *refreshProperties()*. Код старијих верзија, неопходно је својства компоненте освежавати у цикличној функцији *updateHook()*, како би њихов садржај био правовремено ажуриран.

Замена кинематичке компоненте подразумева раскидање веза старе кинематичке компоненте са системом, другим компонентама, и обнављање истих веза са новом кинематичком компонентом. Овакав сценарио је могућ само у случају када је нова кинематичке компонента, која треба да преузме улогу, резидентна у меморији управљачког система. То практично значи да приликом подизања софтвера управљачког система морају бити подигнуте све компоненте које ће у тренутку активирања система или касније ући у састав управљачке конфигурације.

UML дијаграм случајева коришћења у току фазе динамичког реконфигурисања управљачког система дат је на слици 4.50.



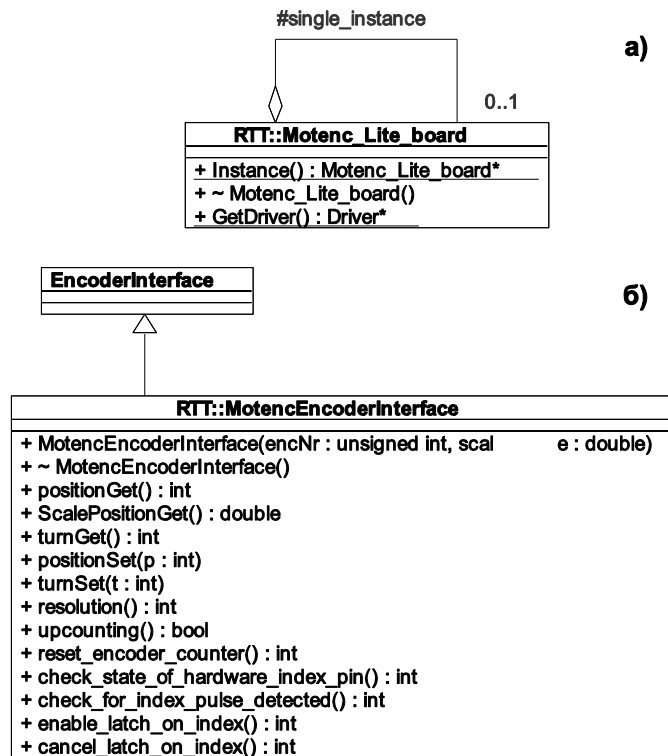
Слика 4.50 Пример динамичког реконфигурисања *real-time* управљачког подсистема

4.2.6 Конфигурисање експерименталног прототипа елементарне конфигурације реконфигурабилног управљачког система на бази софтверских компонената развијених применом *OROCOS*-а

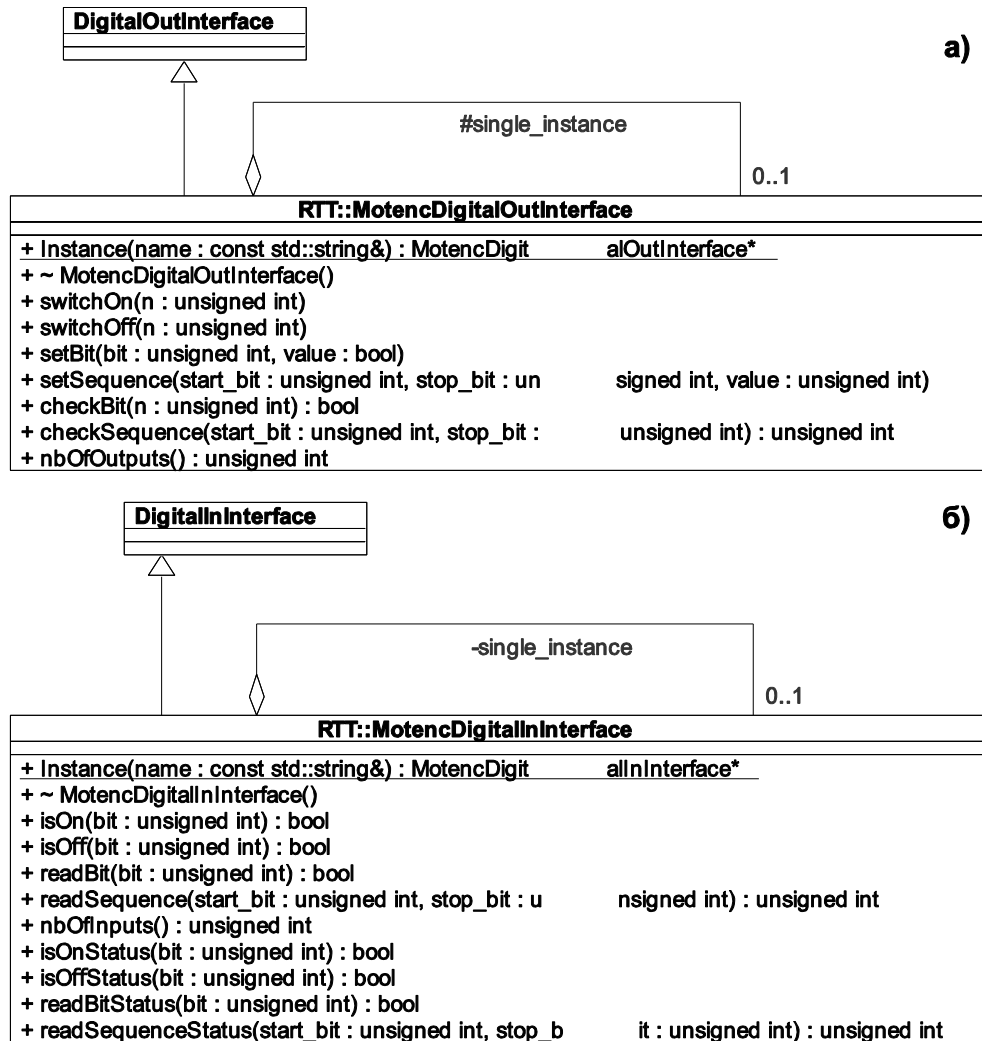
На основу развијених управљачких алгоритама и софтверских компонената, уз коришћење успостављене методологије конфигурисања управљања, приказане у одељку 4.2.5, у Лабораторији за машине алатке и роботе института Лола је конфигурисан експериментални прототип управљачког система. С обзиром на чињеницу да у овој фази развоја реконфигурабилног управљачког система нису све имплементиране

функције постављене у домен *real-time* и *off-line* подсистема, елементарна конфигурација прототипа управљачког система која је тестирана обухватила је следеће софтверске компоненте:

- *L-IRL* компилатор, слика 4.27, који је у овој фази развоја компетиран и расположив за тестирање система;
- Све компоненте интерполатора, које чине функционалну основу компоненте *generatorPos*, слика 4.35, и саставни су део *off-line* подсистема;
- *kinematics* са имплементираним функцијама директне и инверзне кинематике робота Лола 15, слика 4.36;
- *nAxisContrillerPosition*, слика 4.37;
- *robotHardver*, која садржи драјвере спрежених картица *Motenc-Lite*, чија је имплементација садржана класи *RTT::Motenc_Lite_Board*, слика 4.51a, односно у изведеним класама из одговарајућих апстрактних класа *OROCOS*-a:
 - *RTT::MotencEncoderLite*, слика 4.51b,
 - *RTT::MotencDigitalOutInterface*, слика 4.52a,



Слика 4.51 Имплементација класа *RTT::Motenc_Lite_Board* и *RTT::MotencEncoderLite*

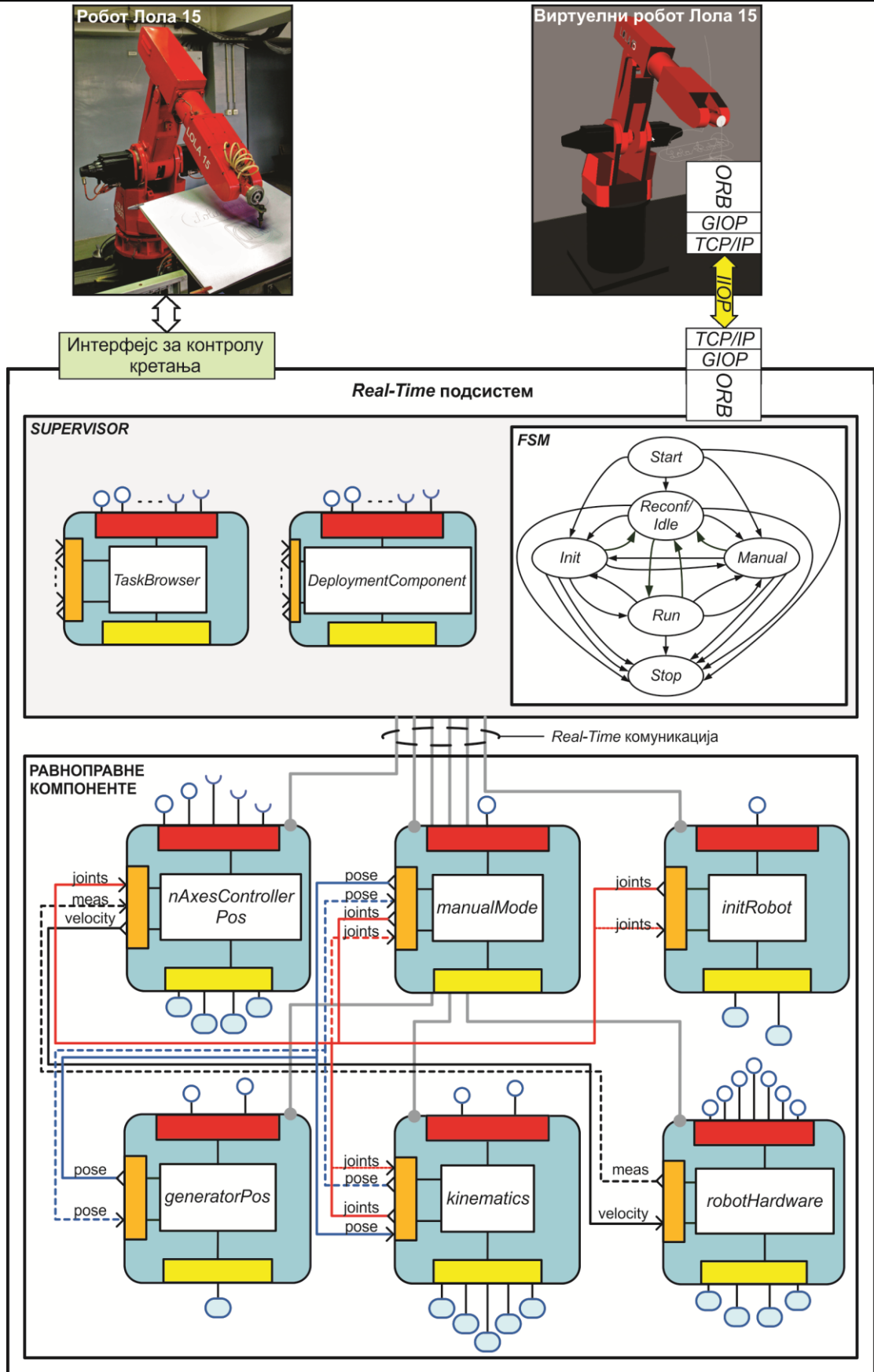


Слика 4.52 Имплементација класа *RTT::MotencDigitalOutInterface* и *RTT::MotencDigitalInInterface*

- *RTT::MotencDigitalInInterface*, слика 4.52б,
- *manualMode*, слика 4.39;
- *initRobot*, слика 4.40.

Шематски приказ веза између компонената управљачког система, конфигурисаног за управљање роботом Лола 15, показан је на слика 4.53. *Real-time* управљачка логика је имплементирана супервизорском машином стања (*FSM*), посредством *OROCOS* софтверских алата. *FSM* ради у реактивном режиму, при чему прелазак управљачког система из једног у друго радно стање подразумева софтверску активацију одговарајућих компонената, односно деактивацију других, уз успостављање, односно раскидање одговарајућих веза.

РАЗВОЈ МЕТОДА ПРОЈЕКТОВАЊА УПРАВЉАЧКОГ СИСТЕМА...



Слика 4.53 Управљачки систем конфигуриран за управљање роботом Лола 15

Прелазак из стања *Start* у стање *Init*, односно *Manual*, захтева активацију компоненте *initRobot*, односно *manualMode*. Улазак у једно од два поменута стања је неопходан по покретању управљачког система због природе инкременталних мерних система робота и неопходности иницијализације. По иницијализацији се може покренути апликативни програм (програм обраде) преласком у стање *Run*, чиме се активира компонента *generatorPos*, која посредством свог излазног порта, слика 4.53, компоненти *kinematics*, слика 4.36, у регуларним временским тренуцима прослеђује информације о позицији робота. При преласку из стања *Run* у стање *Manual*, компонента *generatorPos* постаје неактивна, раскидају се везе са активним компонентама у систему, а њено место заузима компонента *manualMode*, преузимајући функцију генерисања путање. Стање *Init* подразумева да је компонента *generatorPos* замењена компонентом *initRobot*. Ако је у фази иницијалног конфигурисања дефинисано више конфигурација управљачке архитектуре, бира се једна током подизања управљачког система. У току активног статуса *FSM*-а (не сме бити *Stop*) управљачка архитектура се може динамички реконфигурисати, одабиром жељене конфигурације из скупа конфигурација садржаног у *XML* конфигурационој датотеци. Избор нове конфигурације се обавља у терминал прозору компоненте *TaskBrowser*. Реконфигурисање система се обавља у стању *Reconfig/Idle* *FSM*-а, при чему се, на основу информација садржаних у *XML* конфигурационој датотеци, нежељене компоненте деактивирају, а жељене конфигуришу и активирају.

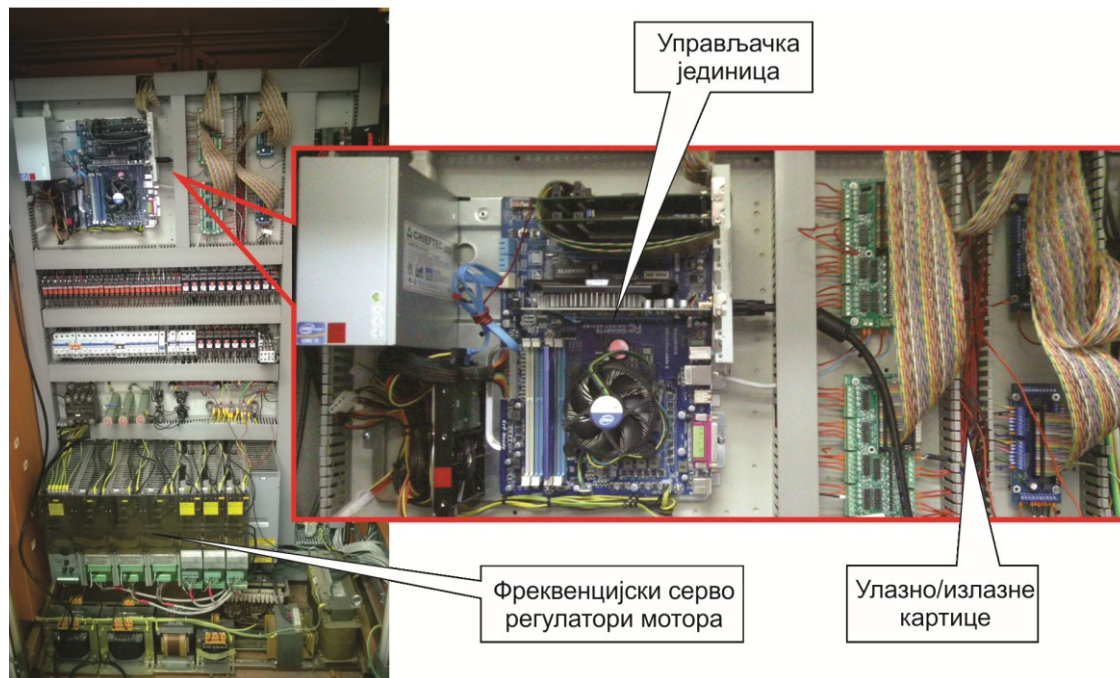
На слици 4.54 приказан је орман управљачке јединице са конфигурисаним рачунарским хардвером. Централни део ормана заузима управљачка јединица развијена на *PC* основи са припадајућим *Motenc-Lite* интерфејс картицама за улазно/излазне операције. У доњем делу ормана се налази енергетски део система задужен за контролу серво мотора у брзинској управљачкој петљи.

Како би се поузданост система подигла на одговарајући ниво, пројектован је и реализован *watchdog*⁴ тајмер, електронски склоп на бази *Cypress*-овог *PSoC* (**P**rogrammable **S**ystem-**o**n-**C**hip) микроконтролера, слика 4.55.

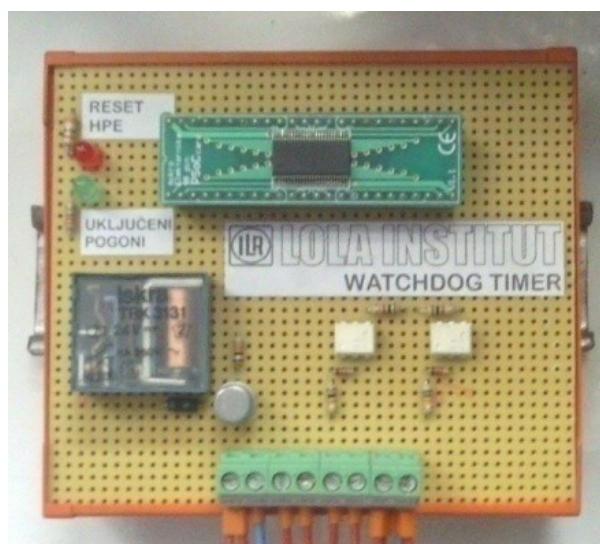
⁴ енг. *watchdog* - израз који је прихваћен у домаћој литератури без превођења

РАЗВОЈ МЕТОДА ПРОЈЕКТОВАЊА УПРАВЉАЧКОГ СИСТЕМА...

Управљачки систем је надграђен софтверским модулom који комуницира са *watchdog* тајмером са периодом од 10 ms, оглашавајући да је систем активан. У случају изостанка сигнала активности, активирају се безбедносне процедуре.



Слика 4.54 Орман управљачке јединице

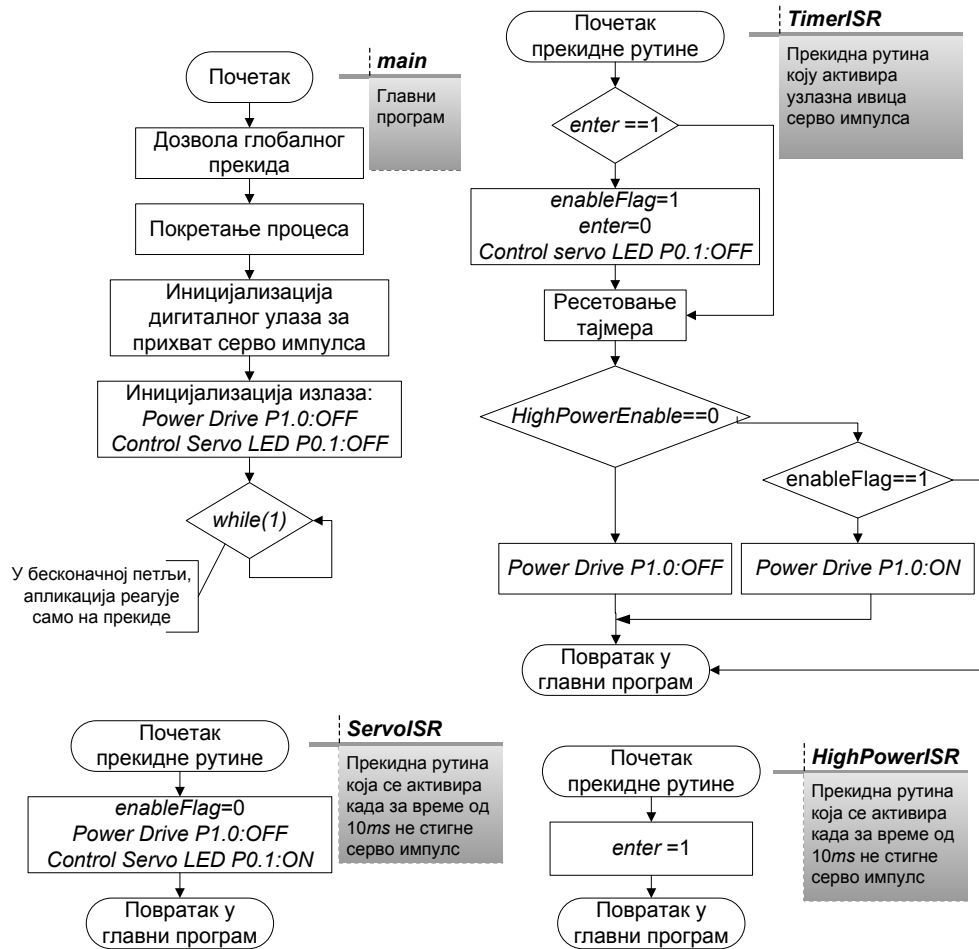


Слика 4.55 *Watchdog* тајмер

Алгоритам који следи *watchdog* тајмер у циљу надгледања активности управљачког система, приказан је на слици 4.56.

Верификација функционалности и перформанси овако конфигурираног управљачког система, коришћењем софтверских компонента управљања

имплементираних у овој фази развоја и интегрисаних у једну елементарну управљачку конфигурацију, приказана је у поглављу 6.



Слика 4.56 Алгоритам функционисања *watchdog* тајмера

5 ВЕРИФИКАЦИЈА ЕКСПЕРИМЕНТАЛНОГ СИСТЕМА УПРАВЉАЊА И ПРОГРАМИРАЊА РЕКОНФИГУРАБИЛНИХ ОБРАДНИХ ЋЕЛИЈА РАЗВИЈЕНОГ НА БАЗИ СОФТВЕРА *EMC2*

Експерименти, који су спроведени током израде ове дисертације, имали су за циљ верификацију резултата до којих се дошло током истраживања. У том циљу су на две идентичне рачунарске платформе, конфигурирана два управљачка система применом метода, описаних у претходном поглављу.

Управљачки систем, развијен на бази готових софтверских и хардверских решења отворене архитектуре применом софтвера *EMC2*, имплементиран је на елементарној обрадној ћелији на бази робота вертикалне зглобне конфигурације Лола 50 са 6 степени слободе, која је за потребе извођења експеримената конфигурирана на Катедри за производно машинство, Машинског факултета у Београду.

Управљачки систем развијен на бази софтвера отворене архитектуре *EMC2*, који омогућава статичко реконфигурисање система, пружио је функционално окружење за извођење експерименталне обраде према иницијално постављеном концепту реконфигурабилности у поглављу 2.

Током експерименталне обраде тестиране су функционалности развијених кинематичких алгоритама, као и могућности емулације машина алатки различитим конфигурацијама робота. Са аспекта безбедности извођења експеримената, посебна пажња је посвећена верификацији и симулацији програма обраде на виртуелним обрадним системима, како би биле избегнуте колизионе ситуације током обраде на реалном роботу.

5.1 План експеримента верификације система управљања и програмирања на бази *EMC2* софтвера

Експериментална верификација функционалности реконфигурабилног управљачког система подразумева примену успостављене методологије конфигурисања обрадних ћелија на бази робота, као и обраду погодних одабраних радних предмета. За моделирање радних предмета и генерисање

програма обраде могу се користити различити савремени стандардни CAD/CAM системи, као што је *PTC Creo 2.0* који је коришћен у овој дисертацији [90]. Основна идеја је да се програмирање обраде и сама обрада роботима обавља на идентичан начин као и на 5-осним машинама алаткама, чији рад емулирају обрадне ћелије на бази робота у овој дисертацији. Пре саме обраде је извршена елементарна калибрација и одређене су референтне позиције за све разматране емулиране варијанте 5-осних машина алатки. С обзиром да работи имају мању тачност и већу попустљивост у односу на машине алатке, за израду радних предмета су коришћени мекши материјали, као што су стиродур и материјали високе густине на бази полиуретана који се веома често користе за израду модела. Употреба стиродура за извођење ове врсте експеримената је посебно погодна због мањих сила резања које се јављају током обраде и минималне могућности појаве нежељених ефеката услед вибрација. Експерименти су обухватили 3-осне и 5-осне обраде аналитички дефинисаних и слободних просторних површина ниже и средње класе тачности, као и сложених естетских и функционалних површина, у циљу верификације успостављених кинематичких модела разматраних конфигурација обрадних ћелија на бази робота, табела 5.1.

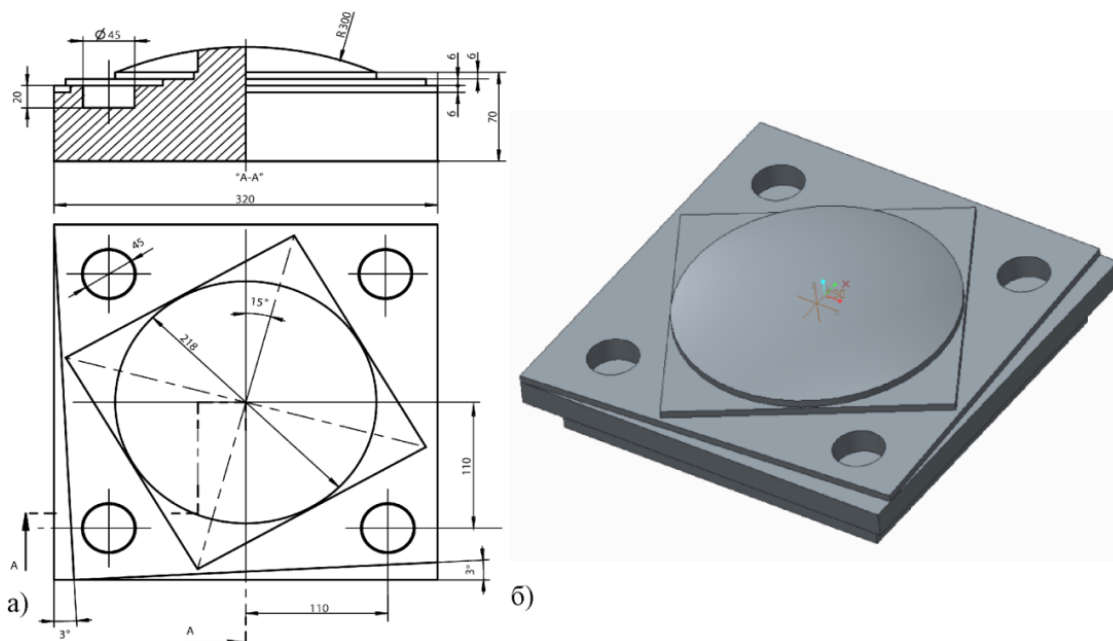
5.2 Експерименти на роботској обрадној ћелији која емулира 5-осне вертикалне глодалице

Експерименти на роботској обрадној ћелији која емулира 5-осне вертикалне глодалице (X, Y, Z, A, B) у конфигурацијама к3.1, слика 3.3, и к3.3, слика 3.5, обухватили су обраду три пробна тест дела који се користе за испитивање радне тачности машина алатки и који укључују 3-осну и 5-осну обраду. У циљу демонстрације могућности обраде сложених функционалних површина, обрађени су примери лопатице турбине и импелера. За приказ могућности примене разматране обрадне ћелије за обраду сложених естетских површина, већих габарита, ниже и средње класе тачности, обрађени су и примери радних предмета анфаса и профила људске главе, грчке античке реплике и средњовековног грба.

Табела 5.1 План експеримента за обраду радних предмета

Конфигурације обрадне ћелије - Радни предмети Тип обраде Алати: Вретенасто равно (VR) Вретенасто лоптасто (VL)		емулиране 5-осне верт. или хориз. глодалице							
		к3.1	к3.2	к3.3	к3.4	к3.5			
Тест пробни делови	 I тест пробни део	Предобрада: Алат (VR):	3-осна <i>d12, l60/30</i>		X	X	X	X	
		Завршна обрада: Алат (VR):	5-осна <i>d12, l60/30</i>						
	 II тест пробни део	Предобрада: Алат (VR):		X	X		3-осна <i>d12, l60/30</i>	X	X
		Завршна обрада: Алат (VL):					3- и 5-осна <i>d12, l60/30</i>		
 III тест пробни део	Предобрада: Алат (VL):	3-осна <i>d11, l140/80</i>		X	X	X	X	X	
	Завршна обрада: Алат (VL):	5-осна <i>d11, l140/80</i>							
 VI тест пробни део	Предобрада: Алат (VR):		X	X	X	3-осна <i>d12, l60/30</i>		X	
	Завршна обрада: Алат (VL):					3- и 5-осна <i>d12, l60/30</i>			
Сложене естетске површине	 Анфас	Предобрада: Алат (VL):	3-осна <i>d11, l140/80</i>	3-осна <i>d11, l140/80</i>		X	X	X	
		Завршна обрада: Алат (VL):	3-осна <i>d11, l140/80</i>	3-осна <i>d11, l140/80</i>					
	 Профил	Предобрада: Алат (VL):	3-осна <i>d11, l140/80</i>		X	X	X	X	
		Завршна обрада: Алат (VL):	3-осна <i>d11, l140/80</i>						
	 Античка реплика	Предобрада: Алат (VR):	3-осна <i>d8, l85/35</i>		X	X	X	X	
		Завршна обрада: Алат (VL):	3-осна <i>d8, l85/35</i>						
 Средњевековни грб	Предобрада: Алат (VL):	3-осна <i>d8, l85/35</i>		X	X	X	X		
	Завршна обрада: Алат (VL):	3-осна <i>d8, l85/35</i>							
Сложене функционалне површине	 Импелер	Предобрада: Алат (VL):	3-осна <i>d8, l85/35</i>		X	X	X	X	
		Завршна обрада: Алат (VL):	3-осна <i>d8, l85/35</i>						
	 Лопатица	Предобрада: Алат (VL):		X	X		5-осна <i>d8, l85/35</i>	X	X
Завршна обрада: Алат (VL):						5-осна <i>d8, l85/35</i>			
 Вишестрана обрада	Предобрада: Алат (VR):		X	X	X	3-осна <i>d12, l60/30</i>	3-осна <i>d12, l60/30</i>		
	Завршна обрада: Алат (VL):					5-осна <i>d12, l60/30</i>	5-осна <i>d11, l140/80</i>		

Први пробни део је пројектован слично стандардном *ISO* тест пробном делу како би се тестирале могућности обраде аналитички описивих контура и површина, 3-осном и 5-осном обрадом, слика 5.1.

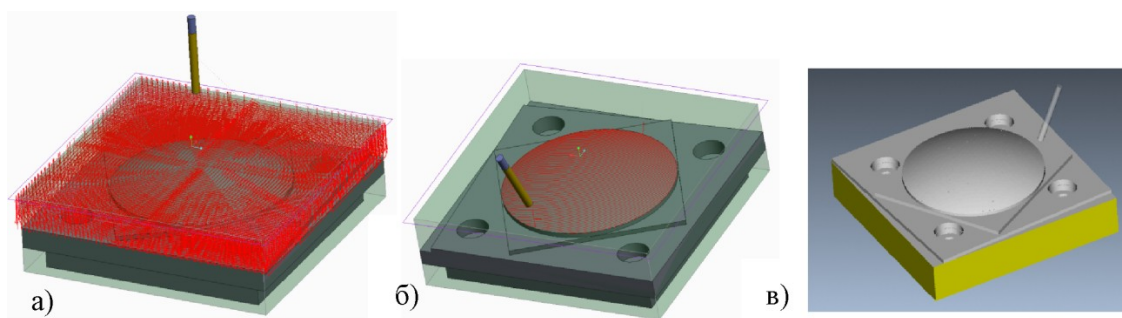


Слика 5.1 Цртеж и *CAD* модел првог пробног дела

На основу цртежа са слике 5.1а, моделиран је *CAD* модел пробног дела, слика 5.1б, за потребе програмирања у *CAD/CAM* систему *PTC Creo 2.0*. У *CAM* систему је прво учитан добијени референтни *CAD* модел пробног дела, изабран координатни систем за програмирање $\{W\}$ на радном предмету и дефинисан одговарајући припремак. Иако у општем случају координатни системи $\{M\}$ и $\{W\}$ не морају бити паралелни, у овој фази развоја система је предвиђено да буду паралелни. У наставку процедуре програмирања изабран је тип машине, 5-осна глодалица, након чега је уследило дефинисање операција за обраду.

У склопу сваке операције потребно је одредити нулту тачку, сигурносну (енг. *Retract*) раван, толеранцију обраде, стратегију за обраду и одговарајући алат. По завршетку дефинисања сваке операције, генерисана је и симулирана путања алата која је сачувана као *CL* датотека (*.ncl), и следећој фази постпроцесирана сагласно стандарду *ISO6983* у формат *G*-кôда за емулирану 5-осну вертикалну глодалицу. На слици 5.2а је показана симулација путање алата за 3-осну предобраду пробног дела, док је на слици 5.2б показана

симулација завршне 5-осне обраде испупчене сферне површине. Као завршна верификација у *CAM* систему спроведена је и симулација уклањања материјала (*NC check*) која је показана на слици 5.2в.



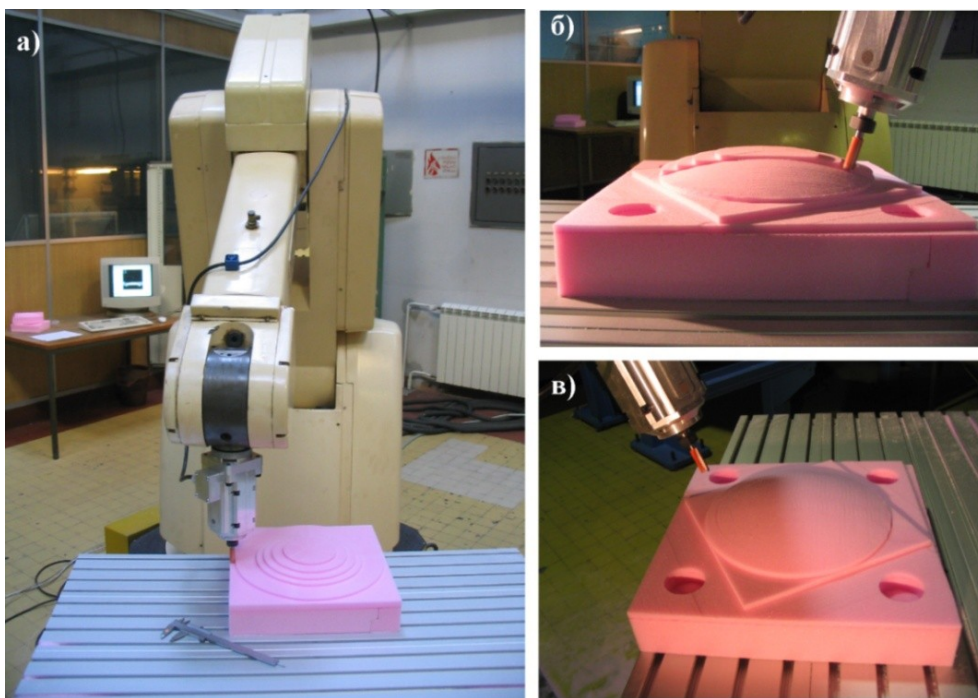
Слика 5.2 Симулације путање алата и уклањања материјала за пример обраде првог пробног дела

После спроведених верификација програма обраде у *CAD/CAM* систему и базирања припремка, програм обраде је пренет у управљачку јединицу обрадне ћелије на бази робота, која је конфигурисана за емулацију 5-осне вертикалне глодалице (X, Y, Z, A, B). Након извршене иницијализације робота, одређене су координате нулте тачке, односно координатног система обрадњака $\{W\}$, слика 5.3.



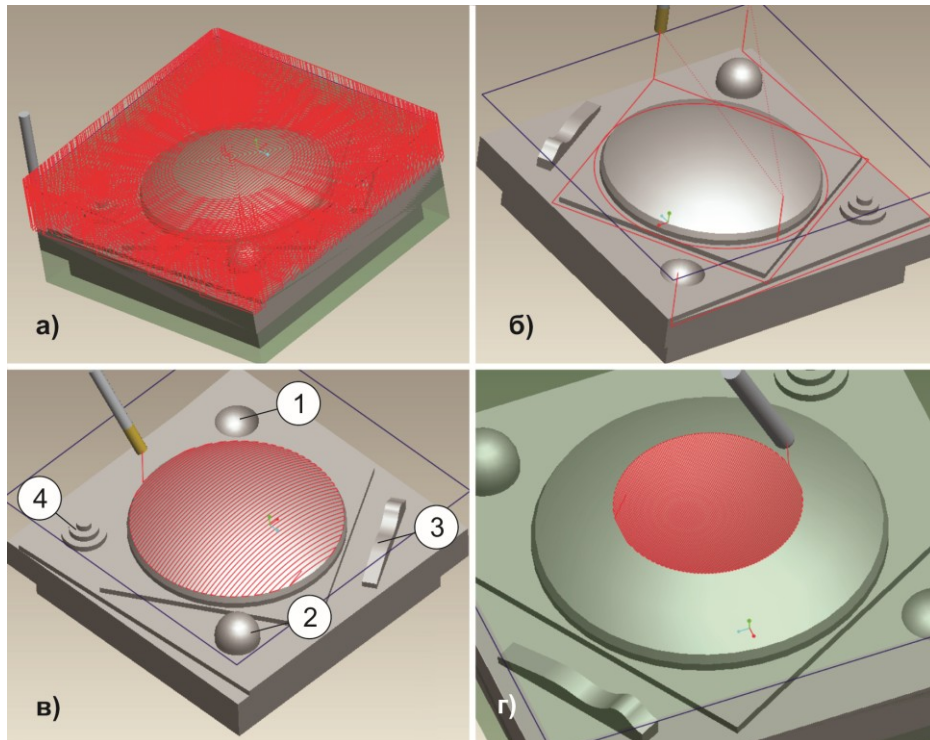
Слика 5.3 Одређивање нулте тачке

Обрада првог пробног дела на емулираној 5-осној вертикалној глодалици (X, Y, Z, A, B), конфигурација к3.1 са слике 3.3, показана је на слици 5.4. Прво је урађена предобрада 3-осном обрадом, слика 5.4а, а затим и завршна 5-осна обрада испупчене сферне површине, слика 5.4б. За обе обраде је коришћено вретенасто равно глодало пречника 12mm , дужине $60/30\text{mm}$. Обрађени део је показан на слици 5.4в.



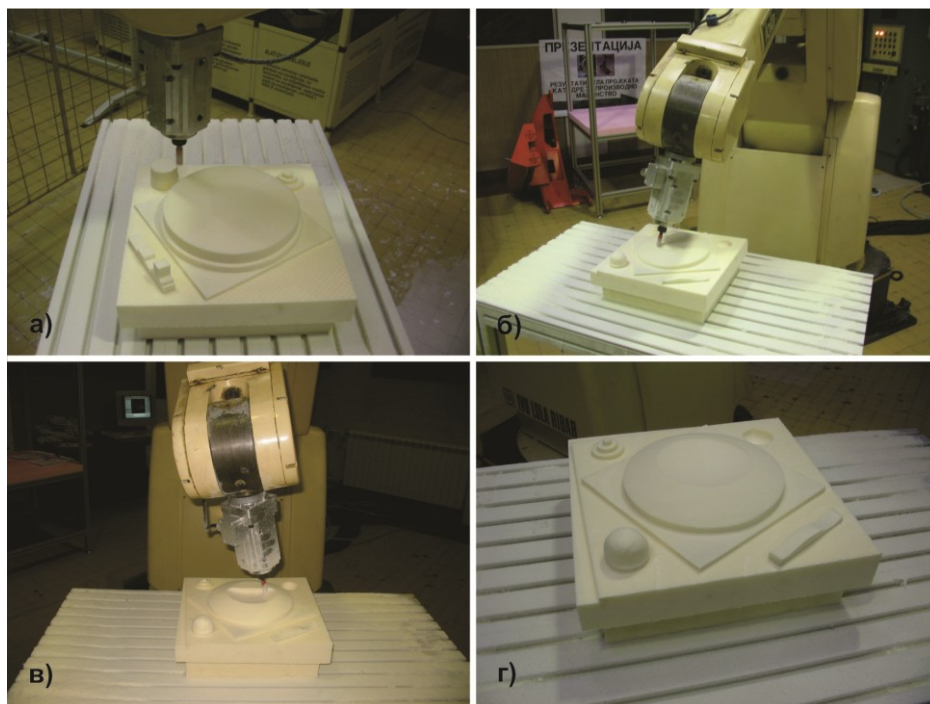
Слика 5.4 Обрада првог пробног дела емулираном 5-осном вертикалном глодалицом (X, Y, Z, A, B) конфигурације к3.1

Други пробни део је пројектован слично стандардним тест пробним деловима за испитивање 5-осних машина алатки који, за разлику од претходног дела, садржи већи број сложенијих геометријских примитива. На слици 5.5а је показана симулација путање алата за 3-осну предобраду другог пробног дела, док је на слици 5.5б показана симулација 3-осне контурне обраде кружном и линеарном интерполацијом. На сликама 5.5в и 5.5г је показана завршна 5-осна обрада испупчене и удубљене сферне површине. Завршна обрада је обухватила и четири типска геометријска елемента: обраду мале удубљене, слика 5.5в ①, и мале испупчене калоте, слика 5.5в ②, таласасте површине, слика 5.5в ③ и степенастих цилиндара, слика 5.5в ④, 3-осном обрадом.



Слика 5.5 Симулације путање алата за пример обраде другог пробног дела

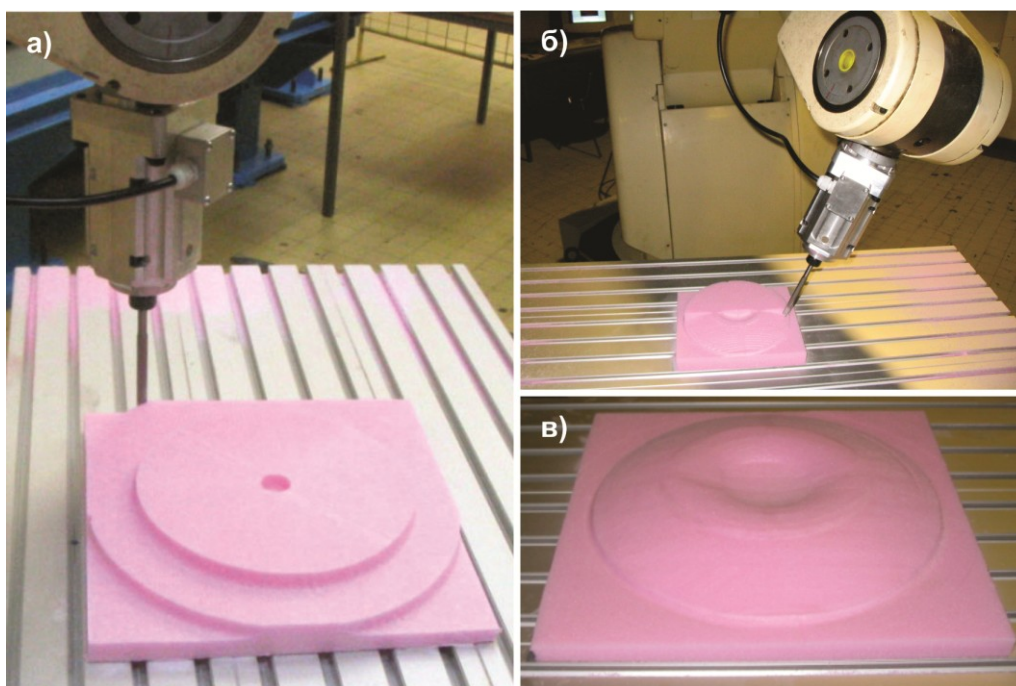
Обрада другог пробног дела на емулираној 5-осној вертикалној глодалици (X, Y, Z, A, B), конфигурација к3.3 са слике 3.5, показана је на слици 5.6.



Слика 5.6 Обрада другог пробног дела емулираном 5-осном вертикалном глодалицом (X, Y, Z, A, B) конфигурације к3.3

Прво је урађена предобрада 3-осном обрадом, слика 5.6а, а затим и 3-осне контурне обраде кружном и линеарном интерполацијом. На сликама 5.6б и 5.6в је показана завршна 5-осна обрада велике испупчене и велике удубљене сферне површине. На слици 5.6г је приказан комплетно обрађен други пробни део. За завршну обраду велике удубљене сферне површине, мале удубљене и мале испупчене калоте и таласасте површине коришћено је лоптасто глодало пречника 12mm , дужине $60/30\text{mm}$, а за све остале обраде коришћено је вретенасто равно глодало пречника 12mm , дужине $60/30\text{mm}$.

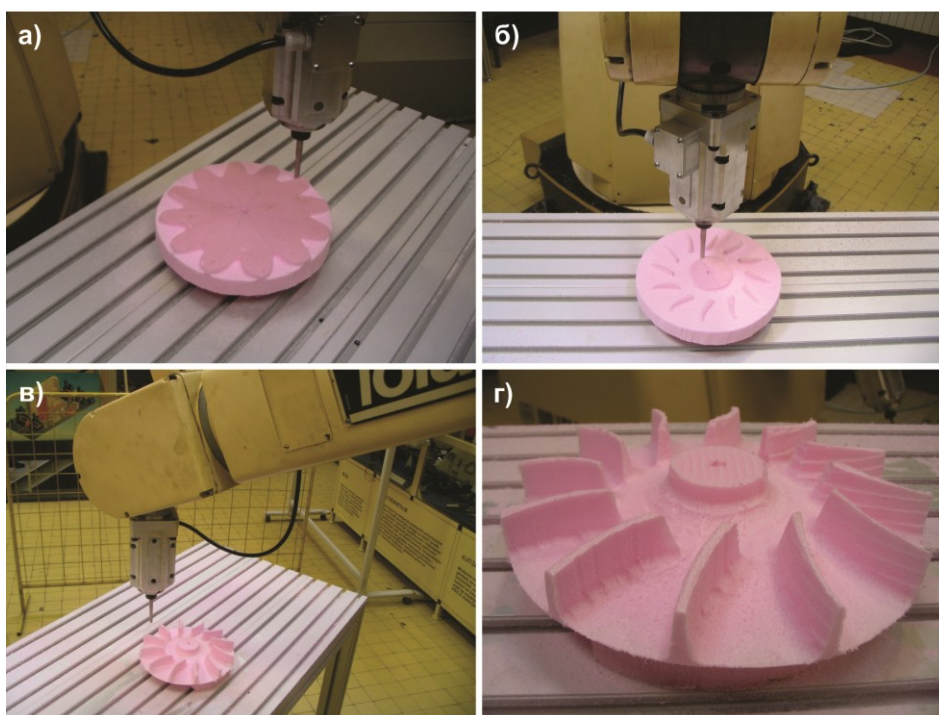
Обрада трећег пробног дела са сложеном аналитичком површином која се састоји од две сферне површине повезане торусном површином је показана на слици 5.7. Обрада је извршена на емулираној 5-осној вертикалној глодалици (X, Y, Z, A, B), конфигурација к3.1 са слике 3.3. Слика 5.7а показује 3-осну предобраду, док је на слици 5.7б показана 5-осна завршна обрада. У оба случаја је коришћено лоптасто глодало дужине $140/80\text{mm}$ и пречника $d = 11\text{mm}$. Овај пример је био веома значајан због провере понашања робота при великим променама оријентације алата на малим растојањима односно при обради заобљења између конкавне и конвексне сфере.



Слика 5.7 Обрада трећег пробног дела емулираном 5-осном вертикалном глодалицом (X, Y, Z, A, B) конфигурације к3.1

Обрада сложених функционалних површина је тестирана на примеру погодно изабраног радног предмета импелера, чија је комплетна обрада остварена 3-осном обрадом, слика 5.8.

Обрадном ћелијом у конфигурацији 5-осне вертикалне глодалице (X, Y, Z, A, B), конфигурације к3.1, извршена је прво предобрада, као што је показано на сликама 5.8а и 5.8б, а затим и завршна обрада, слика 5.8в. На слици 5.8г је показан комплетно обрађен импелер. Алат за израду овог радног предмета је лоптасто глодало дужине 85/35 mm пречника $d = 8\text{ mm}$.

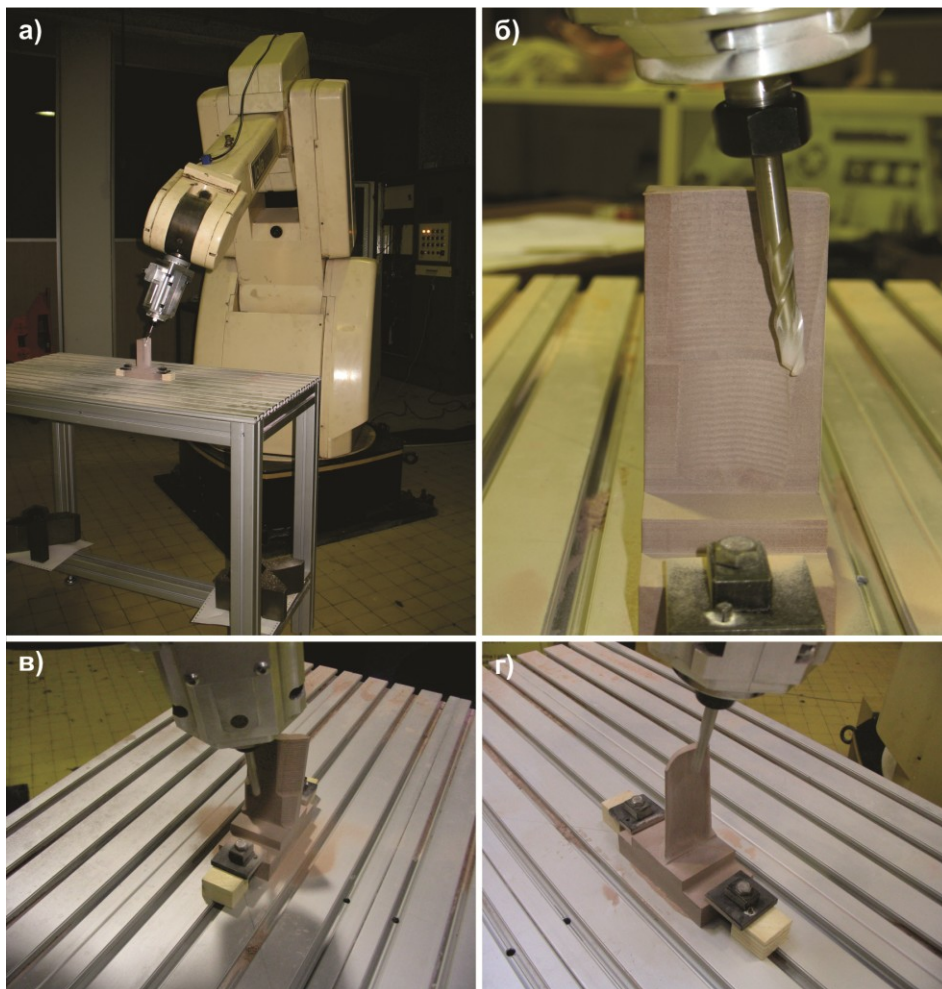


Слика 5.8 Обрада импелера емулираном 5-осном вертикалном глодалицом (X, Y, Z, A, B) конфигурације к3.1

За илустрацију могућности 5-осне обраде сложених функционалних површина, на слици 5.9 је показана израда модела лопатице од материјала високе густине на бази полиуретана (*Sika Block M560*). Обрада је извршена обрадном ћелијом у конфигурацији 5-осне вертикалне глодалице (X, Y, Z, A, B), конфигурације к3.3. Обрада ове лопатице није могућа 3-осном обрадом у једном базирању.

Приликом припреме програма за обраду унапред је задата оријентација алата да би обрада била могућа расположивом дужином алата без колизија, а затим је генерисана путања алата за 5-осну обраду овог радног предмета.

Први од четири пролаза предобраде са једне стране показан је на сликама 5.9а и 5.9б, док је предобрада са друге стране показана на слици 5.9в. Завршна обрада је показана на слици 5.9г. Коришћени алат за овај тест пример је лоптасто глодало дужине 85/35 *mm* пречника $d = 8\text{ mm}$.



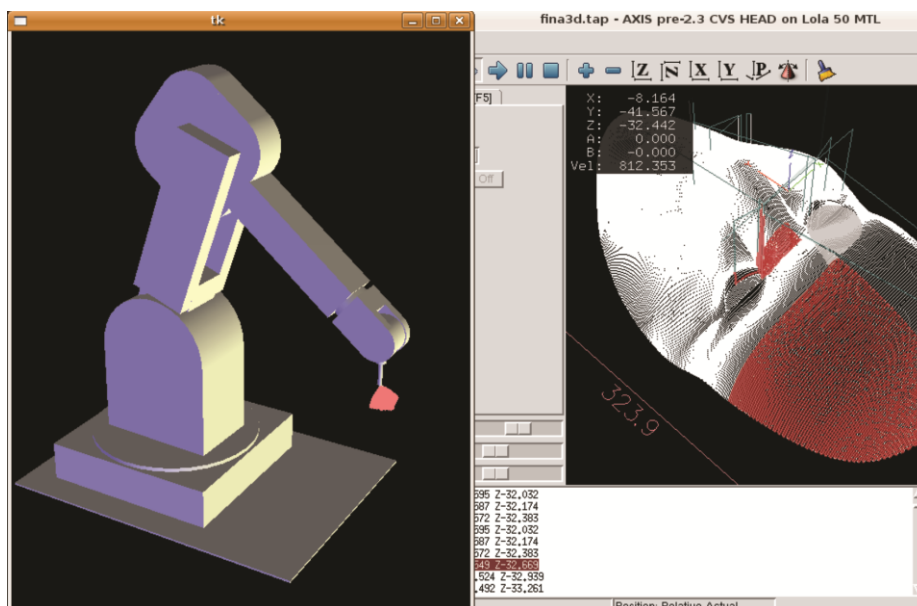
Слика 5.9 Обрада лопатице емулираном 5-осном вертикалном глодалицом (X,Y,Z,A,B) конфигурације к3.3

У циљу демонстрације могућности обраде сложених естетских просторних површина, 3-осном обрадом је обрађено неколико радних предмета: профил и анфас људске главе и рељеф грчке античке реплике.

На примеру обраде анфаса људске главе, показана је и завршна верификација програма обраде на изабраној виртуелној обрадној роботској ћелији у оквиру управљачког система, односно *AXIS* графичког корисничког интерфејса, слика 5.10. Овом верификацијом је омогућен приказ симулираних путања алата добијених интерпретацијом програма обраде у

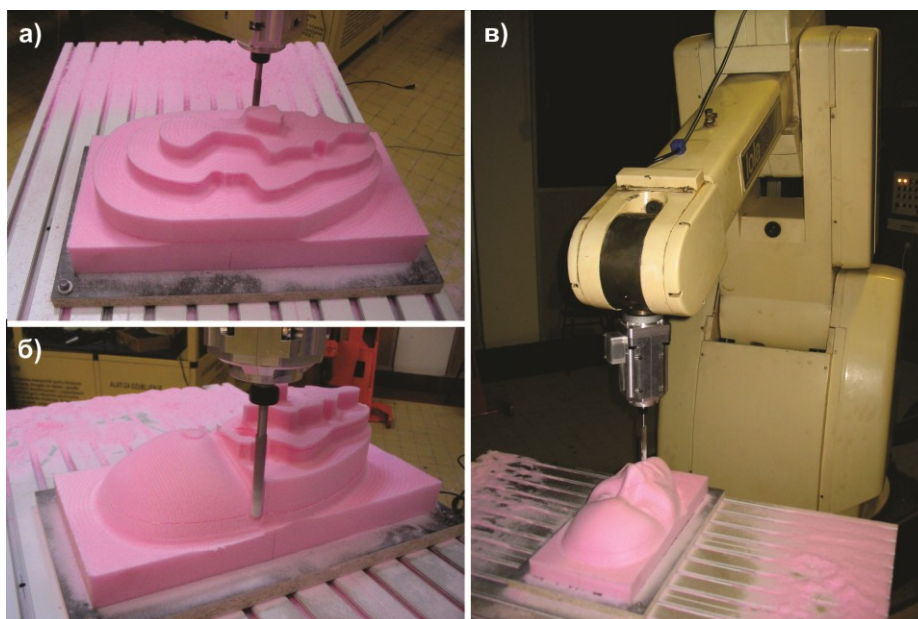
ВЕРИФИКАЦИЈА ЕКСПЕРИМЕНТАЛНОГ СИСТЕМА УПРАВЉАЊА...

AXIS корисничком интерфејсу, односно интерполацијом програмиране путање и кретањем врха алата и сегмената виртуелног робота погоњених идентичним управљачким сигнализима којима се погони и реални робот.



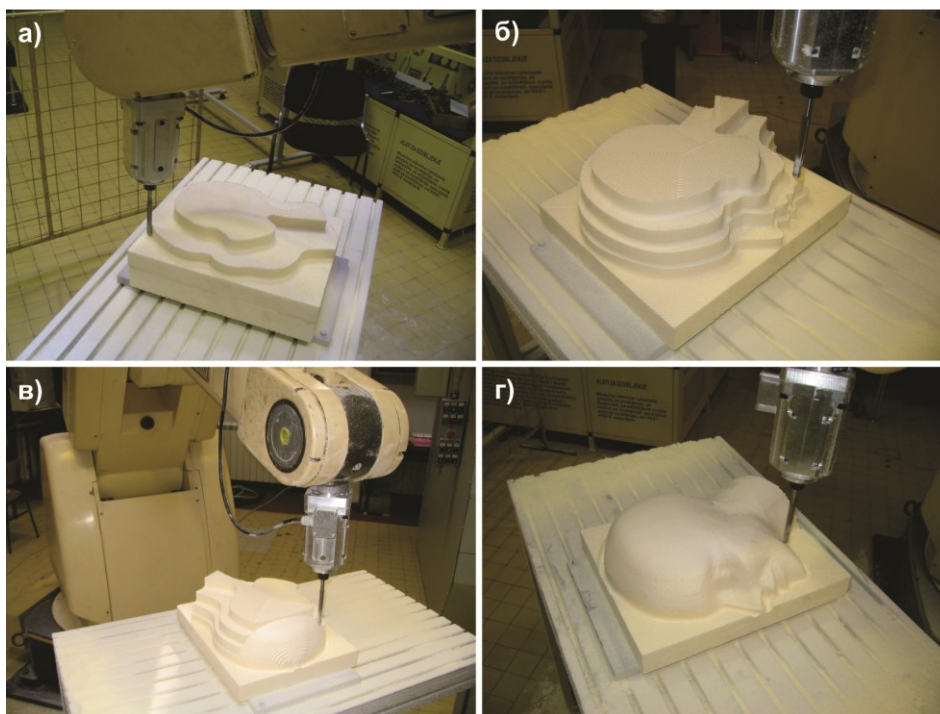
Слика 5.10 Виртуелна обрада анфаса људске главе у управљачком систему

Комплетна обрада анфаса људске главе остварена је 3-осном обрадом и то прво предобрада којом су добијене степенасте површине, слика 5.11а, а затим и завршна обрада, слике 5.11б и 5.11в. За обраду овог дела коришћено је лоптасто глодало дужине $140/80\text{mm}$ и пречника $d = 11\text{mm}$.



Слика 5.11 Обрада анфаса људске главе емулираном 5-осном вертикалном глодалицом (X,Y,Z,A,B) конфигурације к3.1

Као други пример обраде сложених естетских просторних површина, 3-осном обрадом је на сличан начин обрађен и профил људске главе, слика 5.12. На сликама 5.12а и 5.12б је показана предобрада, а на сликама 5.12в и 5.12г завршна обрада. Коришћен је исти алат као и при обради анфаса људске главе.

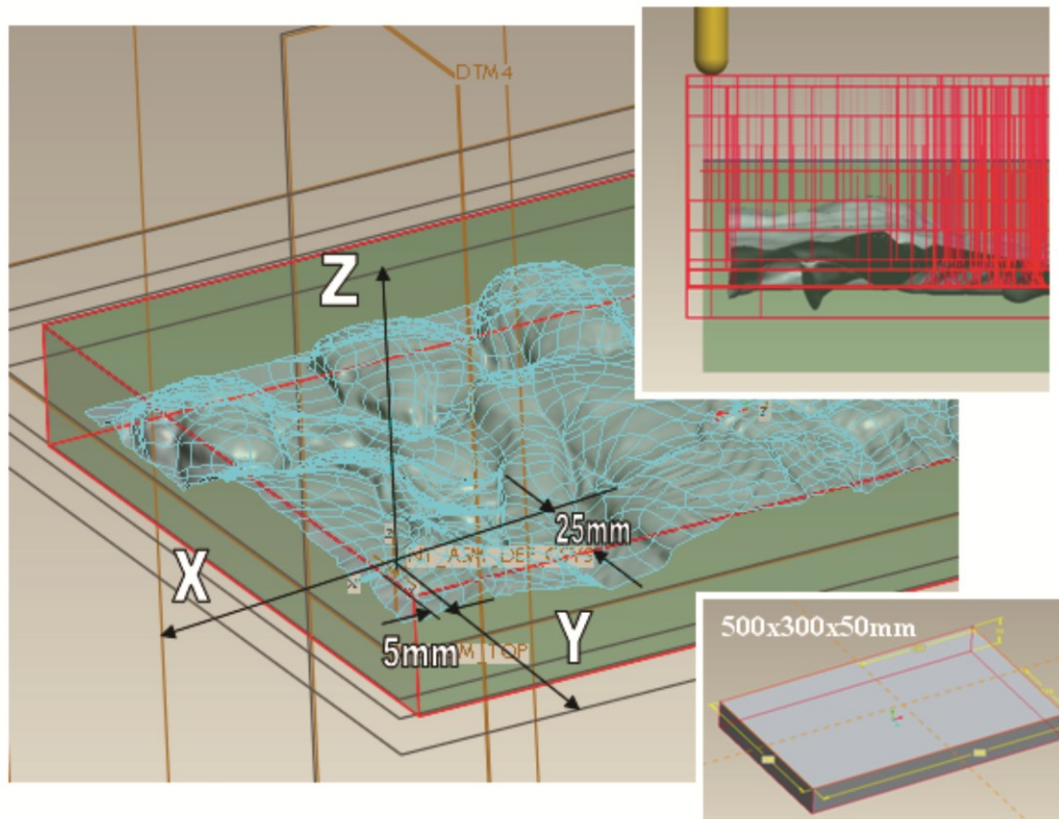


Слика 5.12 Обрада профила људске главе емулираном 5-осном вертикалном глодалицом (X,Y,Z,A,B) конфигурације к3.1

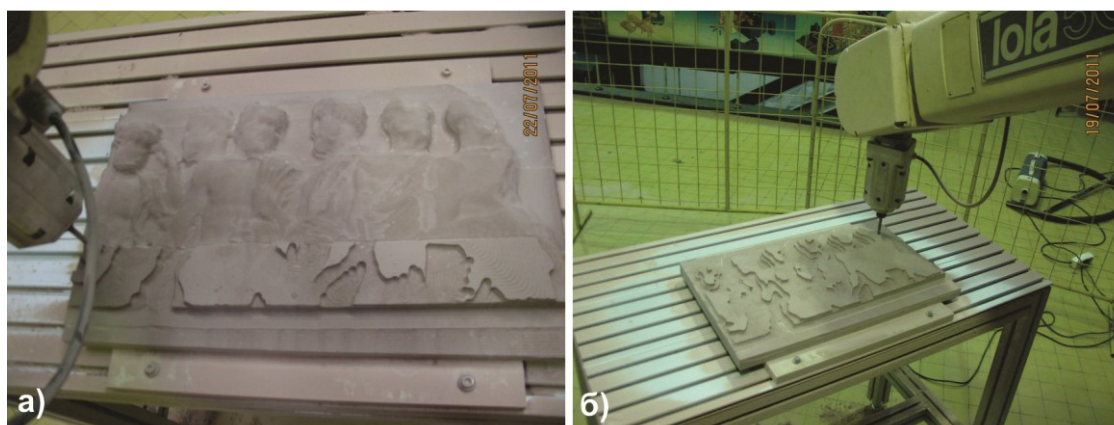
Као пример примене роботске обрадне ћелије за израде реплике објеката културне баштине, остварена је обрада рељефне површине грчке античке реплике на бази добијене *.stl датотеке. На основу *.stl датотеке добијен је модел у CAD/CAM систему и примењена уобичајена процедура програмирања за обраду овог дела. На слици 5.13 је приказан је положај нулте тачке на радном предмету, део генерисане путање алата у равни XZ и припремак за обраду рељефа грчке античке реплике.

По верификацији програма обраде у CAD/CAM систему, симулацијом уклањања материјала (NC Check - VeriCUT), слика 5.14а, приступило се предобради, слика 5.14б, и завршној обради, слика 5.14в, радног предмета израђеног од материјала високе густине на бази полиуретана (Sika Block M560). Детаљ комплетно обрађеног рељефа је приказан на слици 5.14г. За предобраду је коришћено вретенаст равно глодало дужине 85/35mm

пречника $d=8mm$, а за завршну обраду вретенасто лоптасто глодало дужине $85/35mm$ пречника $d=8mm$.

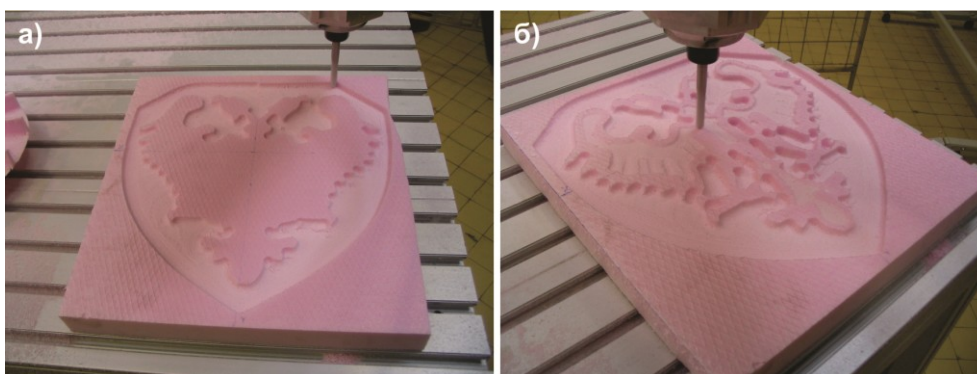


Слика 5.13 Приказ путање алата и припремка за обраду рељефа грчке античке реплика



Слика 5.14 Обрада рељефа грчке античке реплике емулираном 5-осном вертикалном глодалицом (X,Y,Z,A,B) конфигурације к3.1

Српски средњевековни грб је обрађен, како би се показале могућности обраде радних предмета који могу представљати делове позоришне или филмске сценографије, слика 5.15. При томе је коришћена векторизована контура грба, сачувана у **dxf* датотеци, на основу које је генерисан 3D модел у *CAD/CAM* окружењу, а затим спроведена уобичајена процедура програмирања и добијања програма обраде. Слика 5.15а показује предобраду док се на слици 5.15б показује завршна обрада. Алат за израду овог дела је лоптасто глодало дужине 85/35mm пречника $d = 8mm$.



Слика 5.15 Обрада рељефа српског средњевековног грба емулираном 5-осном вертикалном глодалицом (X, Y, Z, A, B) конфигурације к3.1

5.3 Експерименти на роботској обрадној ћелији која емулира 5-осне хоризонталне глодалице

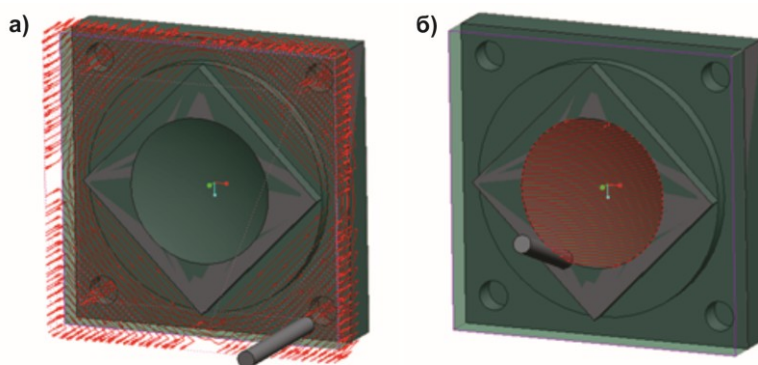
За потребе извођења експеримената на обрадној ћелији која емулира хоризонталне 5-осне глодалице конфигурације (X, Y, Z, A, B), на стандардни интерфејс робота је монтиран угаони интерфејс са моторним вретеном, слика 5.16. Радни предмет је, сагласно захтевима обраде, монтиран на радни сто са угаоником.

Експерименти на роботској обрадној ћелији која емулира 5-осне хоризонталне глодалице (X, Y, Z, A, B) у конфигурацијама к3.2, слика 3.4, и к3.4, слика 3.6, обухватили су обраду једног пробна тест дела који се користе за испитивање радне тачности машина алатки, као и обраду анфаса људске главе у циљу демонстрације могућности обраде сложених естетских површина.



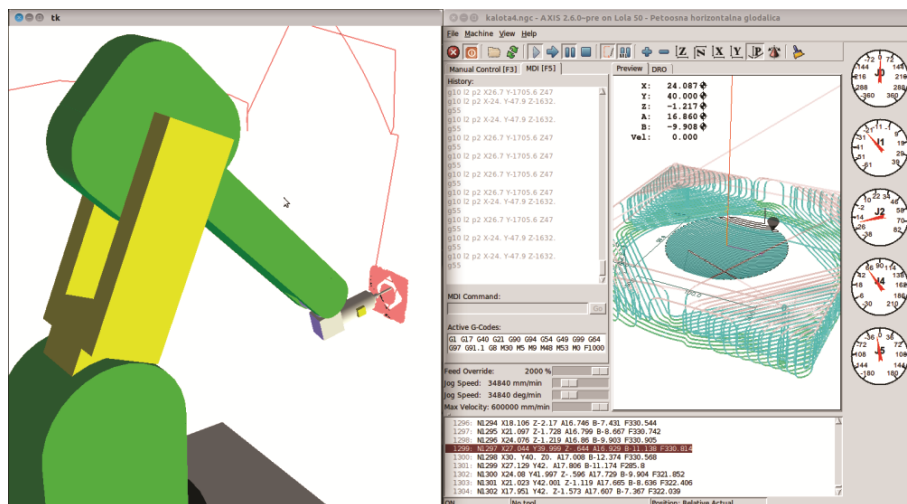
Слика 5.16. Емулација хоризонталне глодалице конфигурације (X,Y,Z,A,B)

Програмирање првог тест пробног дела за обраду на емулираној 5-осној хоризонталној глодалици (X,Y,Z,A,B) је обављено на уобичајени начин у *CAD/CAM* окружењу. Симулације генерисаних путања алата за 3-осну предобраду и завршну 5-осну обраду удубљене сферне површине су показане на слици 5.17.



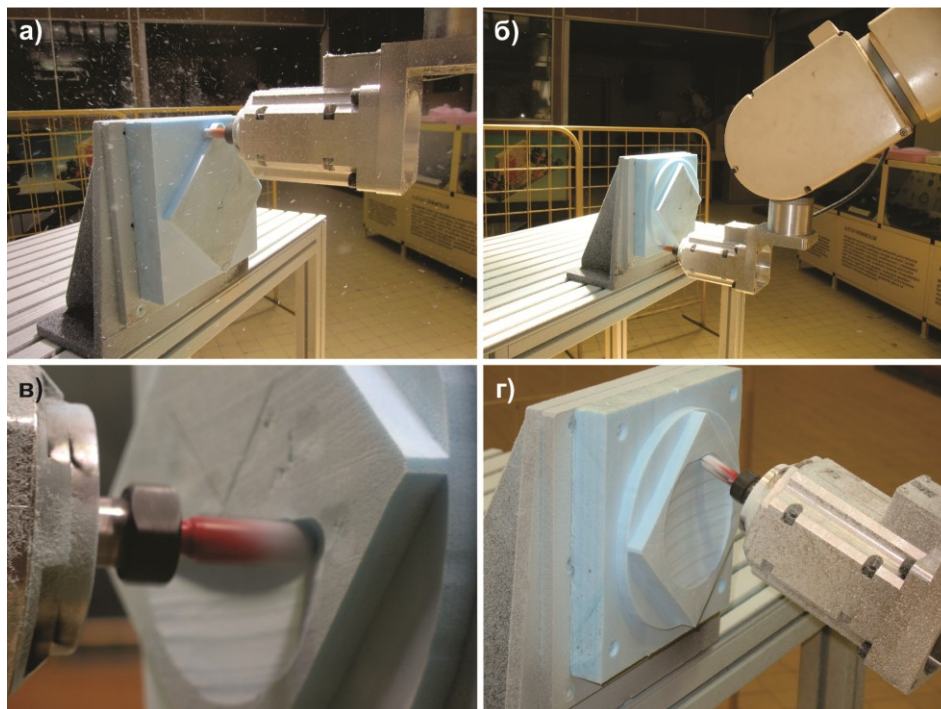
Слика 5.17 Симулације путање алата за пример обраде пробног дела на емулираној 5-осној хоризонталној глодалици (X,Y,Z,A,B)

Верификација програма обраде у управљачком систему, уз приказ симулираних путања алата добијених интерпретацијом програма обраде у *AXIS* корисничком интерфејсу, односно интерполацијом програмиране путање и кретањем врха алата и сегмената виртуелног робота, приказана је на слици 5.18.



Слика 5.18 Симулација обраде пробног дела у управњачком систему

Обрада првог тест пробног дела на емулираној 5-осној хоризонталној глодалици (X, Y, Z, A, B), конфигурација к3.4 са слике 3.6, показана је на слици 5.19. Прво је урађена предобрада 3-осном обрадом, слике 5.19а и 5.19б, а затим и завршна 5-осна обрада удубљене сферне површине, слике 5.19в и 5.19г. За предобраду је коришћено вретенасто равно глодало дужине $60/30\text{mm}$ пречника $d=12\text{mm}$, а за завршну обраду лоптасто глодало истих димензија.



Слика 5.19. Обрада пробног дела на емулираној 5-осној хоризонталној глодалици (X, Y, Z, A, B) у конфигурацији к3.4

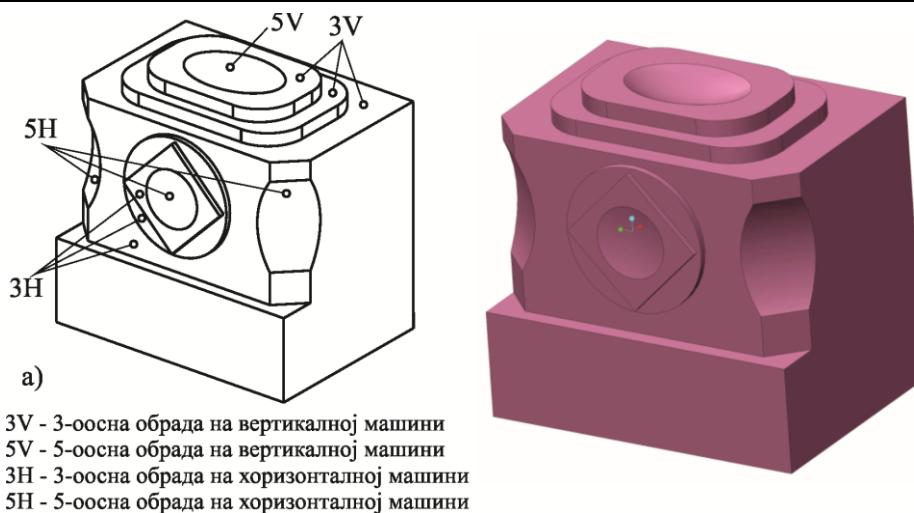
Могућност обраде слободних естетских површина на емулираној 5-осној хоризонталној глодалици (X, Y, Z, A, B) у конфигурацији к3.2, слика 3.4, је демонстрирана на радном предмету у облику анфаса људске главе, слика 5.20, као што је то показано и на примеру емулиране вертикалне 5-осне глодалице, слике 5.10 и 5.11. Алат за израду овог дела је било лоптасто глодало дужине $140/80\text{mm}$ и пречника $d = 11\text{mm}$.



Слика 5.20. Обрада анфаса људске главе на емулираној 5-осној хоризонталној глодалици (X, Y, Z, A, B) у конфигурацији к3.2

5.4 Експерименти примене роботске обрадне ћелије у вишестраној обради

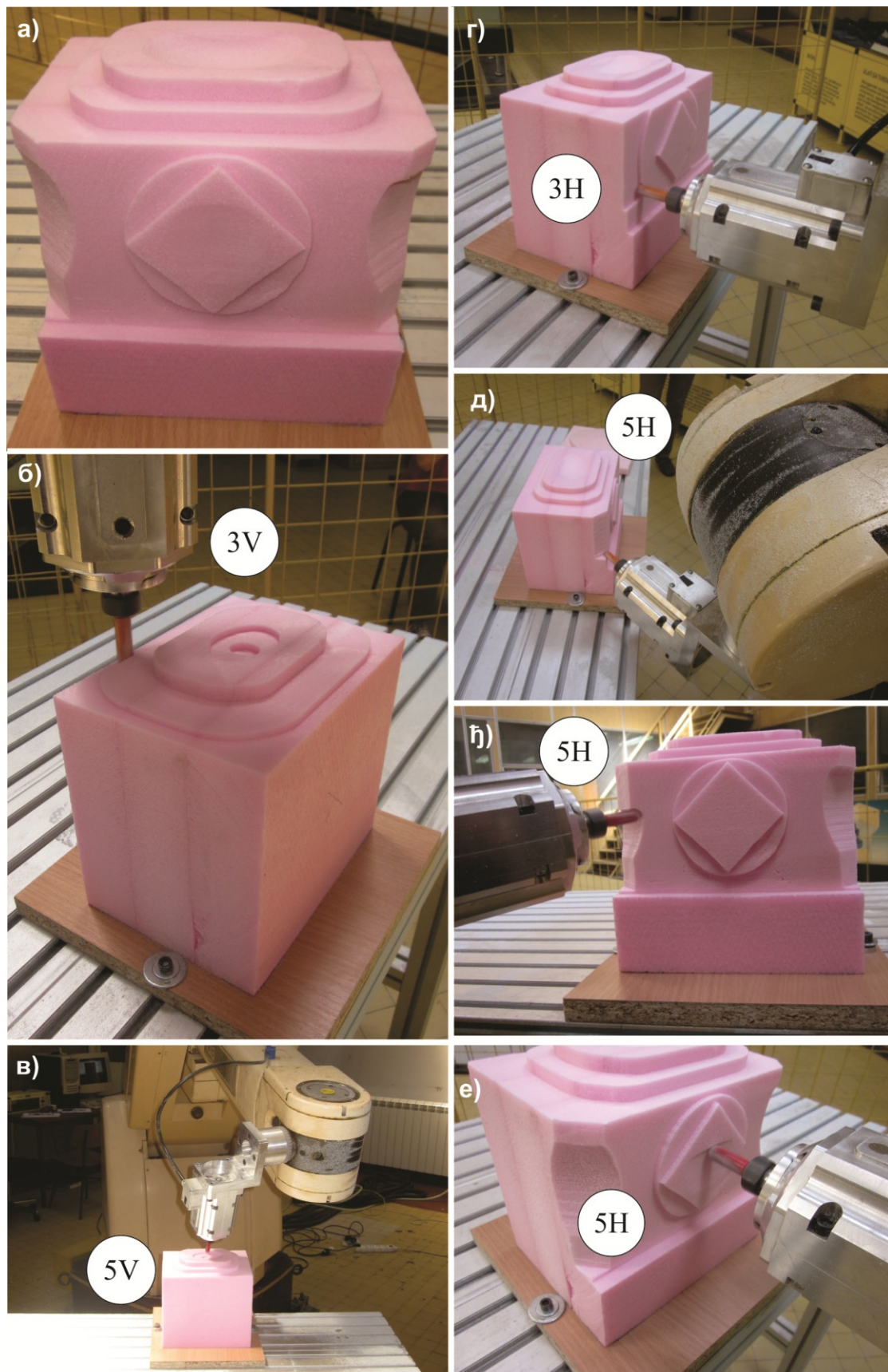
У циљу експерименталне верификације реконфигурисања робота без физичких интервенција, које омогућава развијени систем управљања и програмирања на бази *EMC2* софтвера, пројектован је и обрађен тест пробни део који захтева вишестрану обраду у једном базирању. Слика 5.21 приказује *CAD* модел тест пробног дела за вишестрану обраду са назнаком који се делови геометрије обрађују 3-оосно, који 5-оосно и на којој конфигурацији машине, хоризонталној или вертикалној.



Слика 5.21 CAD модел тест радног предмета за вишестрану обраду

Изглед завршно обрађеног пробног дела од стиродура после обраде показан је на слици 5.22а, док су површине обрађене на следећи начин:

- 3-осном обрадом 3V горње површине пробног дела, слика 5.22б, на емулираној 5-осној вертикалној глодалици конфигурације (X, Y, Z, A, C) конфигурација к3.5;
- 5-осном обрадом 5V удубљења оивиченог елипсоидом, слика 5.22в, на емулираној 5-осној вертикалној глодалици конфигурације (X, Y, Z, A, C) конфигурација к3.5;
- 3-осном обрадом 3H и 5-осном обрадом 5H предње стране пробног дела, слика 5.22г, пројектоване по узору на уобичајене пробне делове, на емулираној 5-осној хоризонталној глодалици конфигурације (X, Y, Z, A, B) конфигурација к3.4;
- 5-осна (3+2) обрада 5H ивица са предње стране под углом од 45°, слика 5.22д на емулираној 5-осној хоризонталној глодалици конфигурације (X, Y, Z, A, B) конфигурација к3.4;
- 5-осна обрада 5H делова удубљених сферних површина на обореним ивицама са предње стране, слика 5.22ђ, на емулираној 5-осној хоризонталној глодалици конфигурације (X, Y, Z, A, B) конфигурација к3.4;
- 5-осна обрада 5H калоте на средини предње стране, слика 5.22, на емулираној 5-осној хоризонталној глодалици конфигурације (X, Y, Z, A, B) конфигурација к3.4.



Слика 5.22 Вишестрана обрада радног предмета

5.5 Анализа реконфигурабилног управљачког система роботских ћелија за обраду развијеног на бази *EMC2*

За потребе извођења експеримената примене различитих конфигурација обрадних ћелија на бази робота, показан је већи број обрађених радних предмета који потврђују исправност имплементираних кинематичких алгоритама за разматране конфигурације обрадних ћелија на бази робота и верификују развијени систем управљања и програмирања на бази софтвера отворене архитектуре *EMC2*. Структура *EMC2* софтвера, као и испољене карактеристике током експерименталне верификације имплементираног управљачког система, указују на следеће:

- Платформски независна имплементација омогућава да се *EMC2* може извршавати на различитим оперативним системима и хардверским платформама;
- *EMC2* садржи *G*-кôд интерпретер реализован по *NIST*-овом стандарду *RS-274NGC* са проширењима која омогућавају рад са променљивама и изразима, коришћење неструктурираних петљи, безусловних и условних скокова;
- Корисник има могућност избора између неколико модуларних графичких корисничких интерфејса (*GUI*) са приказом релевантних информација у току рада робота као што су позиција врха алата, позиције унутрашњих координата робота, текућа линија *G*-кôд која се извршава, графички приказ трајекторије врха алата у 3D графичком окружењу;
- Интерполатор путање заснован на *acceleration/deceleration* алгоритму [86] омогућава да се обрада заустави у било ком тренутку, што даје додатну могућност оператеру да спречи евентуалну колизиону ситуацију без хаваријског заустављања робота;
- За комуникацију између *Linux* и *real-time* процеса и међусобну комуникацију између *real-time* модула, *EMC2* користи дељени меморијски простор (*ShM* – ***Shared Memories***), слика 4.3. За спречавање конфликта приликом симултаног приступа програмских нити *ShM*-у,

користе се мутекси⁵. Прекомерна употреба мутекса смањује ефикасност и повећава комплексност система;

- Приликом интерпретирања, *G*-кôд се преводи у међу код који садржи каноничке функције обраде (*CMFs* – *Canonical Machining Functions*) по стандарду који је дефинисао *NIST*. Иако је *G*-кôд интерпретер декларативно конципиран као независан модул, који посредством *CMFs* комуницира са модулом за управљање кретањем (*EMCMOT*), немогуће га је заменити другим интерпретером истих излазних карактеристика без великих импликација на остатак софтверског система. Тиме је практично онемогућена реконфигурабилност система на интерпретерском нивоу.

⁵ енг. mutex (**mutual exclusion**) - *process-wide lock* механизам

6 ЕКСПЕРИМЕНТАЛНА ВЕРИФИКАЦИЈА ФУНКЦИЈА И ПЕРФОРМАНСИ ОСНОВНИХ КОМПОНЕНАТА ЗА РЕАЛИЗАЦИЈУ РЕКОНФИГУРАБИЛНОГ УПРАВЉАЧКОГ СИСТЕМА НА БАЗИ *OROCOS*-а

Експериментална верификација функција и перформанси основних компонената за реализацију управљачког система, који се конфигурише на бази софтверских алата и хардверских технологија отворене архитектуре уз развој алгоритама управљања, започета је у Лабораторији за машине алатке и роботе Лола института. Према одељку 4.2, односно одељцима 4.2.1 до 4.2.4, експериментална верификација се овде односи на тестирање основних компонената реконфигурабилног управљачког, које су имплементирани у току израде ове дисертације. Применом методологије конфигурисања реконфигурабилног управљачког система, дате у одељку 4.2.5, уз коришћење имплементираних основних компонената управљања, конфигуриран је први експериментални прототип управљачког система на *PC* хардверској платформи која укључује (i) микропроцесор *INTEL I5* са радним тактом од *3GHz*, (ii) радну меморију капацитета *4GB* са радним тактом од *1,33 GHz*, (iii) *MOTENC Lite* интерфејс за улазно/излазне операције. Експериментална верификација функција и перформанси основних софтверских компонената у оквиру првог експерименталног прототипа управљачког система је обухватила:

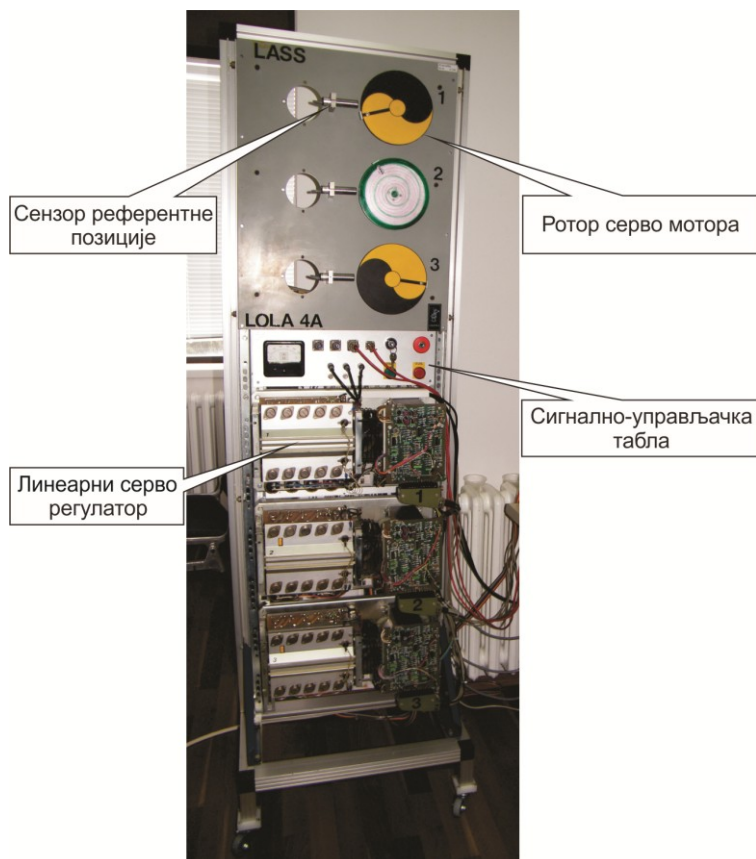
- Развој и имплементацију тест лабораторијског система са 3 аналогна серво погона за потребе тестирања експерименталног управљачког система, односно:
 - Верификацију развијених драјверских модула спрежног хардвера интегрисаних у компоненту *robotHardware*, слика 4.38;
 - Верификацију компоненте *initRobot* која садржи секвенцу иницијализације, слика 4.41;
 - Тестирање функција и перформанси управљачког система у реалном времену;
- Тестирање експерименталног прототипа управљачког система на

расположивом роботу Лола 15, одговарајућим тест програмима, претходно симулираним у 3D виртуелном окружењу.

6.1 Верификација функција и перформанси компонената експерименталног прототипа управљачког система на тест лабораторијском систему са 3 аналогна серво погона

Пре примене на роботу, за потребе иницијалног тестирања функција и перформанси експерименталног прототипа реконфигурабилног управљачког система, развијен је и реализован тест лабораторијски систем са 3 аналогна серво погона, слика 6.1, кога чине:

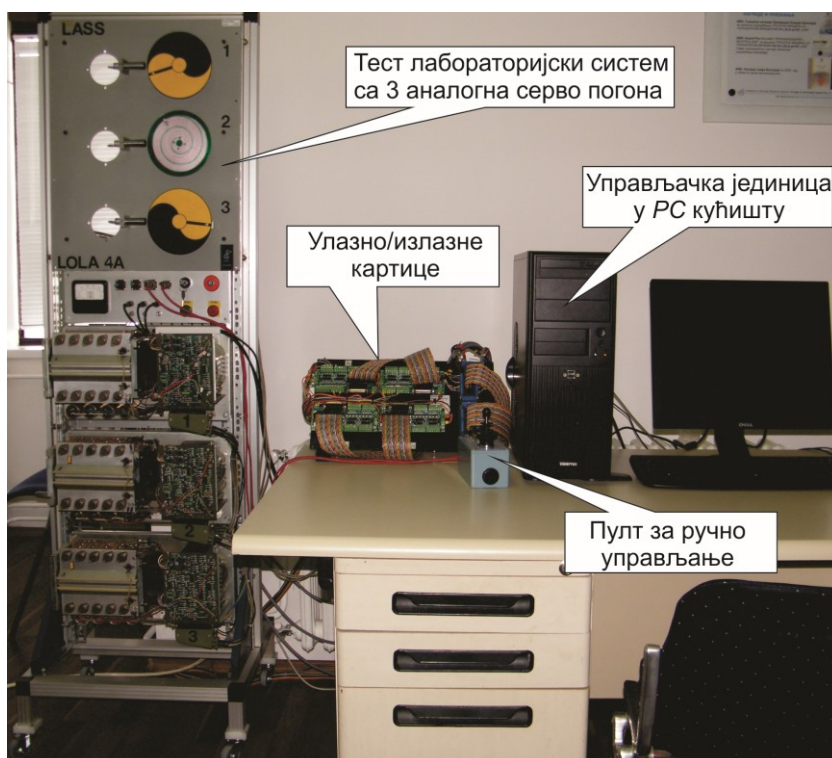
- Три расположива једносмерна серво мотора са перманентним магнетима у статору;
- Три расположива серво регулатора са линеарном регулацијом арматурне струје;
- Три инкрементална оптичка енкодера везана за роторе мотора;



Слика 6.1 Тест лабораторијски систем са 3 аналогна серво погона

- Три индуктивна сензора у зони произвољно изабраних референтних позиција.

Серво петља по брзини је у регулаторима затворена посредством тахогенератора (сензора брзине) на роторима мотора, уграђеним од стране произвођача. Серво петља по позицији се посредством инкременталних оптичких енкодера, интерфејса и драјвера за улазно/излазне операције, затвара унутар компоненте *nAxesContorllerPos*, слика 4.37, експерименталног прототипа управљачког система, слика 6.2.



Слика 6.2 Окружење за тестирање управљачког система

Расположива три серво мотора развијеног тест лабораторијског система су коришћена за тестирања у оквиру којих су вршена *RTP* кретања прве три серво осе робота, при чему су остале осе држане у почетним позицијама. На овај начин је омогућена верификација функција и перформанси оса које доминантно учествују у остваривању задате позиције алата, односно оса које имају снажне погоне и могу утицати на безбедност у каснијим фазама тестирања на роботу. Развијени тест лабораторијски систем је најпре искоришћен за тестирање функција софтверских компонената интегрисаних у први прототип експерименталног управљачког система, а затим и за

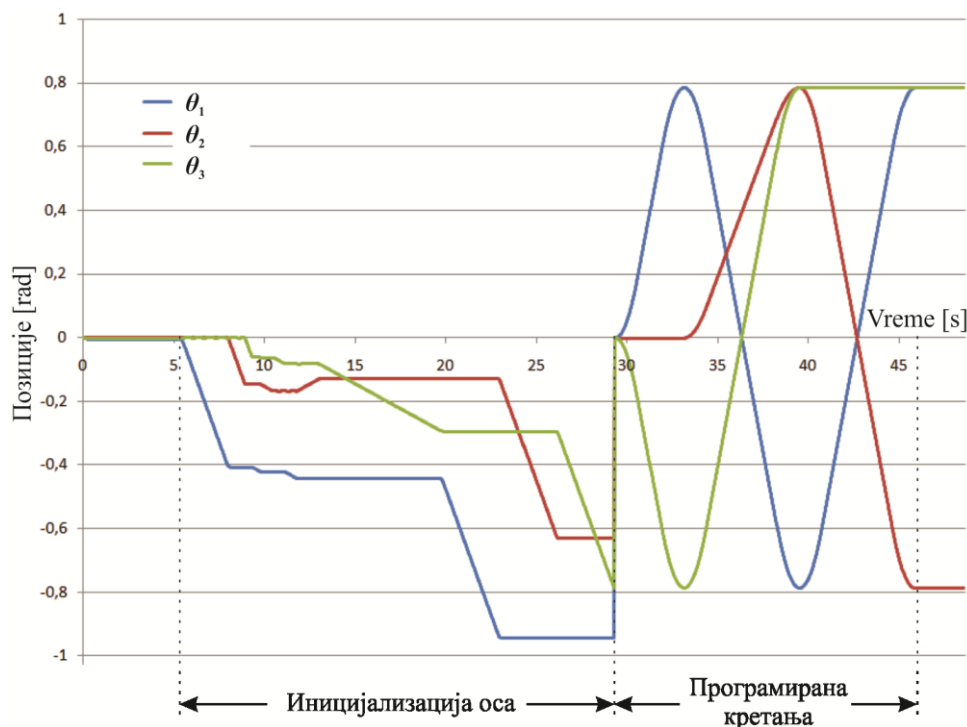
тестирање процедуре иницијализације мерних система, слика 4.41, коришћењем индуктивних сензора. Верификација драјверских модула интерфејса за улазно/излазне операције експерименталног прототипа управљачког система, који су интегрисани у компоненту *robotHardware*, слика 4.38, обављена је на тест лабораторијском систему извођењем *PTP* кретања, програмираним на роботском језику *L-IRL*, слика 6.3, чиме је изузета кинематичка компонента из управљања. Циљ је био да се током тестирања драјверских модула активирају само најнеопходније софтверске компоненте, како би се што ефикасније идентификовали евентуални недостаци у функционисању система.

Позиције мотора током иницијализације оса и извршавања програма, које су посредством енкодера забележене у управљачком систему, слика 6.4,

```

program test:
system_specification "lola15.spc";
seq
  move ptp joint(mainjoint( 45, 0,-45, 0, 0, 0)) speed_ptp:=0.1 acc_ptp:= 0.1 act_rob:"lola_15";
  move ptp joint(mainjoint(-45, 45, 45, 0, 0, 0)) speed_ptp:=0.1 acc_ptp:= 0.1 act_rob:"lola_15";
  move ptp joint(mainjoint( 45, -45, 45, 0, 0, 0)) speed_ptp:=0.1 acc_ptp:= 0.1 act_rob:"lola_15";
endseq
endprogram:
    
```

Слика 6.3 Програм на *L-IRL* језику за тестирање трајекторија три осе, θ_1 до θ_3 , у синхронном *PTP* моду



Слика 6.4 Позиције ротора мотора током тестирања драјверског модула спрежног хардвера за улазно/излазне операције

потврђују правилно функционисање интерфејса за улазно/излазне операције са припадајућим драјверским модулима.

Real-time перформансе експерименталног прототипа управљачког система су добијене мерењем времена извршавања одговарајућих функција софтверских компонената у петљи са више понављања. У табелама 6.1 и 6.2 су приказани резултати мерења.

Табела 6.1 Времена извршавања припремних функција [ms]

функција процес	<i>configureHook</i>	<i>startHook</i>	Прелазак са <i>startHook</i> на <i>updateHook</i>	<i>updateHook</i>
<i>generatorPos</i>	0,5350	0,0100	4,9900	-
<i>nAxesContorllerPos</i>	0,0900	0,0160	0,9840	0,0047
<i>robotHardware</i>	0,1950	0,0400	0,9850	0,0481

Табела 6.2 Времена извршавања функција интерполатора [ms]

функција процес	Израчунавање интерполационих параметара	Интерполациони циклус линеарног кретања	Интерполациони циклус кружног кретања
<i>generatorPos</i>	0,0290	0,0080	0,0110

С обзиром на периодичну активност компонената у управљачком систему, чије су периоде и приоритети (макс. 99) извршавања за изабрану хардверску платформу усвојени као:

- *generatorPos*: приоритет 98, периода 5ms;
- *kinematics*: приоритет 98, периода 5ms;
- *nAxesControllerPos*: приоритет 99, периода 1ms;
- *robotHardware*: приоритет 99, периода 1ms;
- *manualMode*: приоритет 98, периода 10ms;
- *initRobot*: приоритет 98, периода 10ms,

измерена времена извршавања датих функција, које представљају извршни кôд наведених компонената, показују да су одабрани хардверски ресурси у потпуности задовољавајући, односно да остављају могућност паралелног извршавања и других корисничких апликација на истој хардверској платформи.

6.2 Верификација елементарне конфигурације експерименталног прототипа управљачког система

Верификација елементарне конфигурације експерименталног прототипа управљачког система је вршена у лабораторији Лола института, на роботу вертикалне зглобне конфигурације Лола 15 са 6 степени слободе, слика 6.5. Како управљачки систем, поред подсистема за управљање роботом у реалном времену, укључује и *off-line* подсистем са виртуелним 3D окружењем за проверу програма, верификацијом је обухваћена и виртуелна роботска ћелија на којој су извођени тест програми.

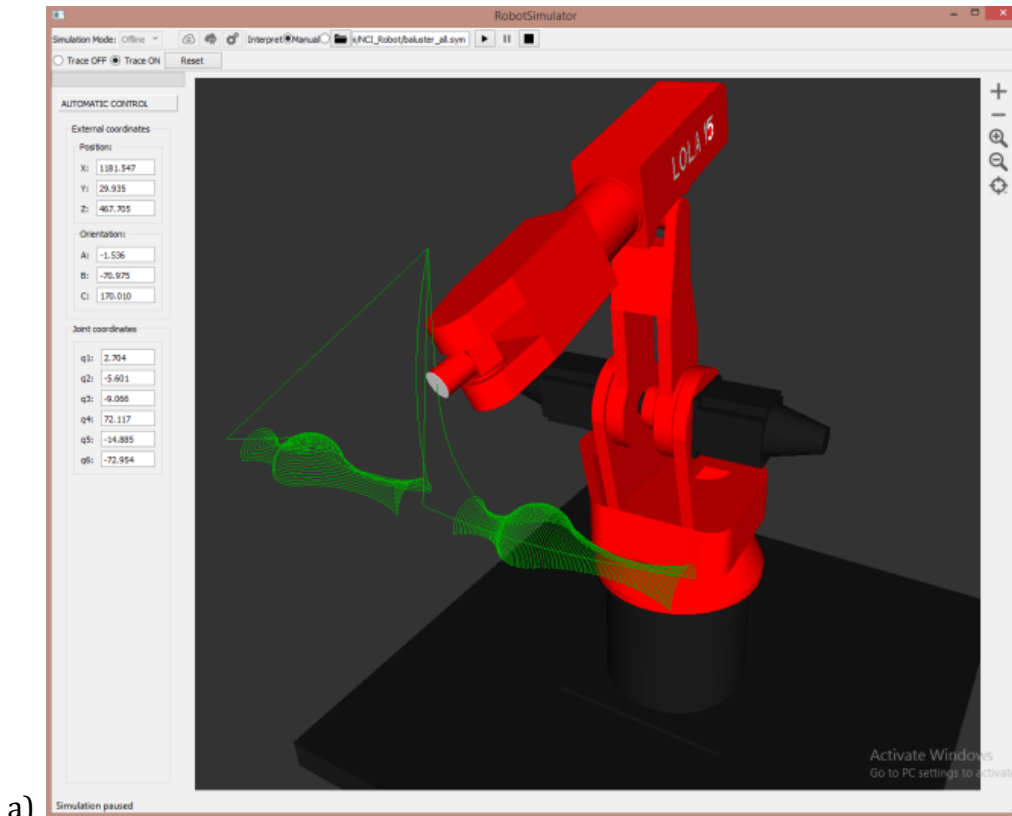


Слика 6.5 Робот вертикалне зглобне конфигурације Лола 15 са 6 степени слободе

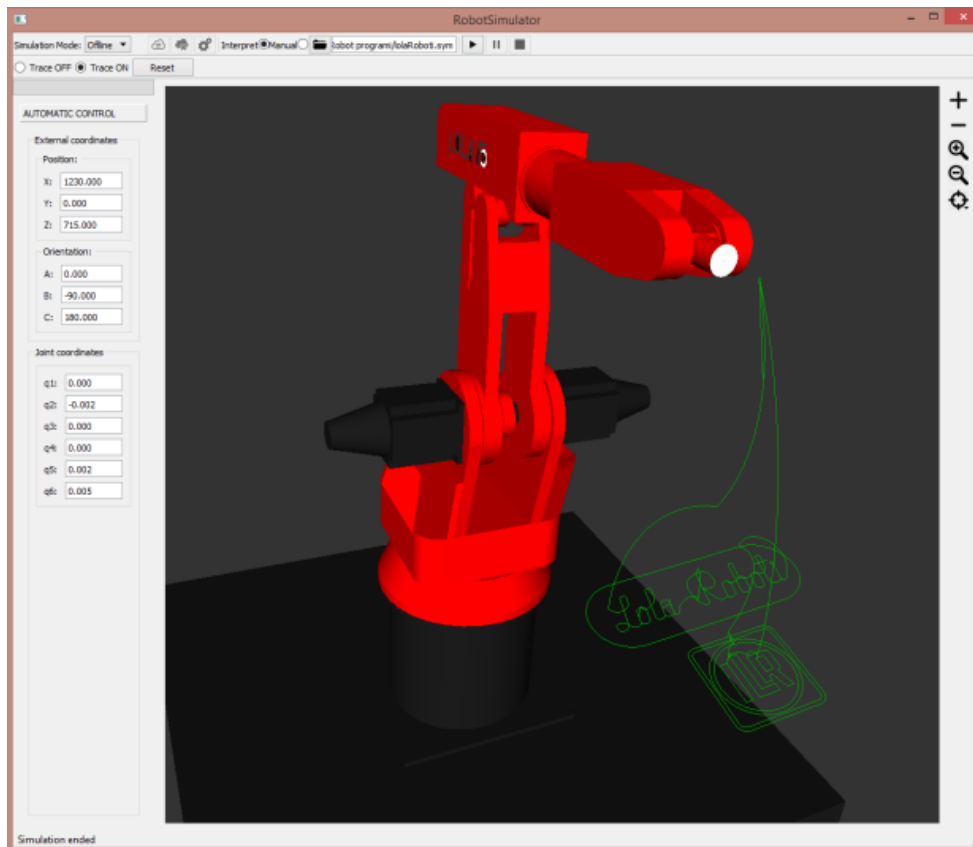
Могућност примене *L-IRL* језика у оквиру имплементираног софтверског модула компилатора, као и расположивост опреме у Лабораторији Лола института, у овој фази развоја су условили конфигурисање експерименталног прототипа управљачког система за управљање роботом Лола 15. Имајући у виду чињеницу да за робот Лола 15 у овој фази развоја нису расположиви обрадна јединица и одговарајући додатни интерфејси, за сложеније случајеве обраде верификација је урађена на *3D* виртуелном роботу, слика 6.6а, док је експериментална провера *real-time* подсистема на роботу рађена једноставним уређајем за писање који симулира моторно вретено. За потребе тестирања, једноставнији програми обраде су писани ручно, а сложенији су генерисани у *MasterCAM*-у посредством постпроцесора за генерисање *L-IRL* програма, који је развијен током рада на овој дисертацији. Пре преношења програма на експерименталну управљачку јединицу, вршено је тестирање на *3D* виртуелном роботу, слика 6.6б, које омогућава уочавање нерегуларних кретања робта изван опсега унутрашњих координата, као и визуелну детекцију колизија између сегмената робота, алата и радног стола.

На основу добијених резултата мерења на тест лабораторијском систему, односно констатацији да прототип експерименталног управљачког система на задовољавајући начин користи рачунарске ресурсе, управљачки софтвер је проширен виртуелним системом за надзор рада роботске ћелије, слика 6.7. У прозору виртуелног система за надзор се може пратити кретање виртуелног робота, које је одређено сигнаlima добијеним са енкодера реалног робота, тренутна позиција реалног робота у виду спољашњих и унутрашњих координата, као и путања врха алата.

Како би биле сагледане могућности робота, односно експерименталног управљачког система, за извођење 5-осне обраде, једноставним уређајем за писање, који симулира моторно вретено, је извођено сложено програмирано кретање у косој равни, слика 6.8.

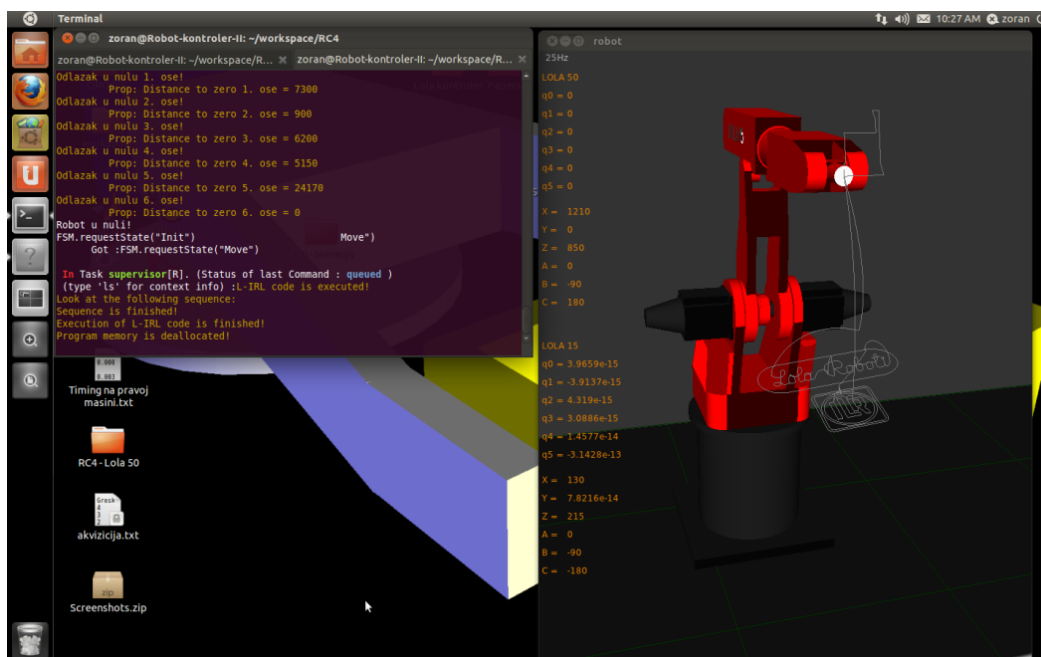


a)



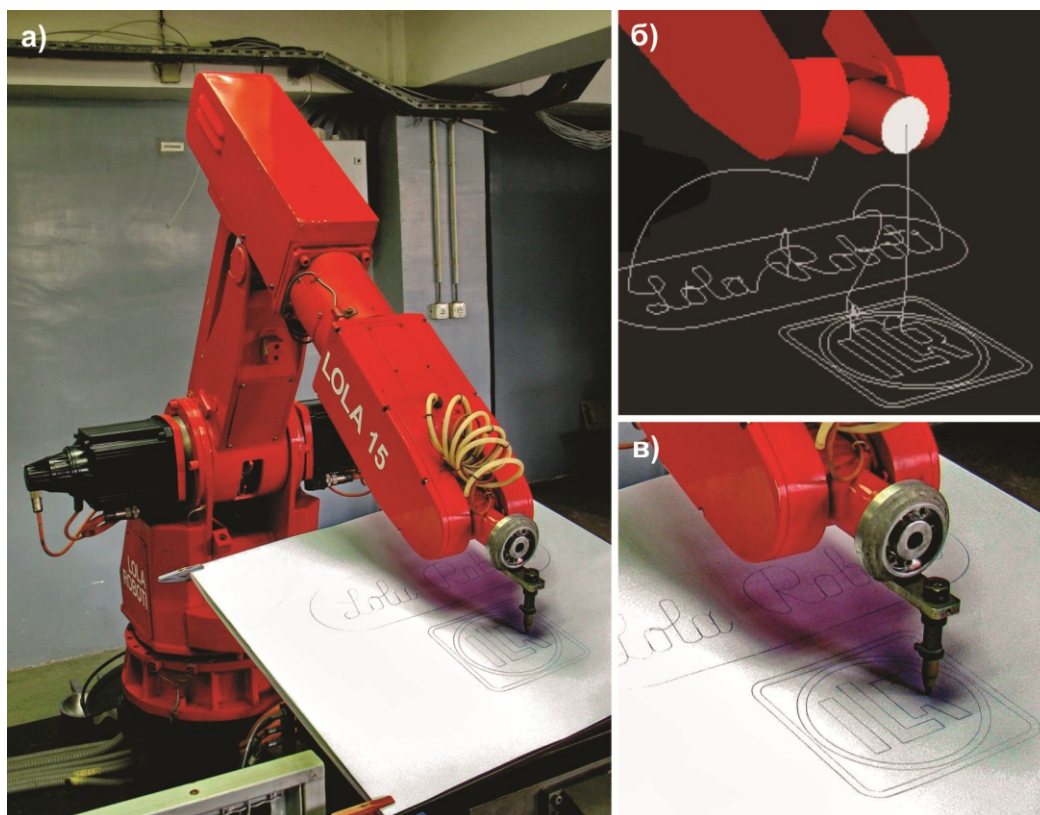
б)

Слика 6.6 Виртуелно 3D окружење за тестирање програма



Слика 6.7 Систем за контролу и праћење рада робота по извршеном задатку

Слике 6.8а и 6.8в приказују робота током извршавања програма, док је на слици 6.8б приказан детаљ виртуелног система за праћење рада роботске хелије.



Слика 6.8 Експеримент на роботу Лола 15

Извођењем тест програма, најпре на тест лабораторијском систему са 3 аналогна серво погона, а затим и на роботу, виртуелном и реалном, потврђена могућност управљачког система да по задатом програму синхроно управља осама робота, испуњавајући при томе захтеве за временским детерминизмом активних софтверских компонената.

6.2.1 Анализа реконфигурабилног управљачког система на бази *OROCOS*-а

За потребе експерименталне верификације развијених софтверских компонената реконфигурабилног управљачког система на бази *OROCOS*-а, предложеног у овој дисертацији, конфигурисан је експериментални прототип елементарне конфигурације управљачког система, који је подвргнут тестирању. Експериментима су показане функције и перформансе развијених софтверских компонената, чиме је потврђен постављен метод развоја реконфигурабилног управљачког система на бази софтверских алата и технологија отворене архитектуре уз развој алгоритама и софтверских функција управљања. Структура софтверског система управљања, као и испољене карактеристике током експерименталне верификације конфигурисаног експерименталног прототипа управљачког система, указују на следеће:

- Платформски независна имплементација софтверских компонената управљања, која је омогућена применом *OROCOS*-а као извршног окружења са неутралним *real-time* апликативним интерфејсом, омогућава реализацију управљачког система на најразличитијим хардверским платформама и оперативним системима;
- Апликативни интерфејс реализованих *OROCOS* софтверских компонената, посредством *CORBA*-е омогућава динамичко укључивање и искључивање компонената у систем, као и дистрибуцију компонената на више хардверских платформи повезаних одговарајућим мрежним протоколом;
- Реконфигурабилност је омогућена на свим хијерархијским нивоима управљања: (*i*) на нивоу компилатора се поред *G*-кôда и *STEP-NC*-а за

програмирање може одабрати роботски језик, што је показано током експерименталне верификације, (ii) предвиђено је да се на нивоу интерполатора се могу одабрати само оне врсте кретања које су кориснику потребне, умањујући редундантност и цену система за захтевану групу задатака, (iii) на нивоу драјверских модула интерфејса за улазно/излазне операције, поставком метода развоја драјвера као саставног дела компоненте управљачког система, применом објектно-оријентисаног начина пројектовања, дата је могућност примене најразличитијих савремених погонских система;

- Развијено окружење за конфигурисање виртуелних роботских ћелија као аутономног софтверског система са апликативним интерфејсом који интегрише *CORBA*-у, омогућава његову примену за: (i) *off-line* програмирања за симулацију програма обраде, (ii) праћење рада робота у реалном времену на самом управљачком систему, (iii) удаљено праћење посредством мреже и одговарајућег мрежног протокола. Виртуелни систем за симулацију обрадних ћелија на бази робота се може користити, као независна апликација, за едукације у обради роботима;
- Иако у овој фази нису имплементиране све пројектом предвиђене компоненте реконфигурабилног управљачког система, укључујући и *G*-кôд компилатор за кога је преостало довршити генератор *P*-кôда, реализација компонентата које су коришћене за конфигурисање експерименталног прототипа управљачког система је захтевала обиман програмерски рад, као и развој и имплементацију додатних ресурса за верификацију и тестирање развијеног прототипа.

7 ЗАКЉУЧАК

Интензивна истраживања у области развоја реконфигурабилних обрадних система на бази робота се данас односе решавања две групе проблема, комплексности програмирања и незадовољавајуће крутости структуре робота. Пратећи ове трендове, Катедра за производно машинство Машинског факултета у Београду и Лола институт су покренули вишегодишња истраживања, која се односе на:

- Развој реконфигурабилног система управљања и програмирања робота у G -кôду;
- Повећање тачности обраде услед статичке попустљивости робота при обради.

Чињенице да сложеност програмирања робота за обраду, као и „нормално затворена“ архитектура комерцијално доступних управљачких система за роботе представљају највеће препреке за ширу примену робота у обради, у овој дисертацији су биле основни мотив за развој управљачког система отворене архитектуре реконфигурабилних роботских ћелија за обраду са могућношћу програмирања обраде у G -кôду.

Идеја да се постојећи *CAD/CAM* системи користе за програмирање робота у G -кôду за вишеосну обраду резањем, представља основу за приступ развоју управљачког система робота, који омогућава емулацију хоризонталних и вертикалних 5-осних машина алатки, роботима вертикалне зглобне конфигурације. Структура реконфигурабилне роботске ћелије за обраду, која емулира хоризонталну или вертикалну 5-осну машину алатку, одређује дефинише структуру софтверских и хардверских модула реконфигурабилног управљачког система као и њихове међусобне везе. Овакав приступ је подразумевао идентификацију, класификацију и кинематичку анализу скупа изабраних конфигурација ћелије, која поред основне кинематичке структуре робота вертикалне зглобне конфигурације са 5 или 6 степени слободе, укључује и модуле додатних обртних и/или транслаторних оса.

При конципирању реконфигурабилног система управљања роботским ћелијама за обраду пошло се од елементарне роботске ћелије која укључује

један робот вертикалне зглобне конфигурације са 6 степени слободе и фиксни радни сто, при чему у току обраде сва позициона и помоћна кретања изводи робот. Оваквим приступом је омогућено да стандардизовани робот са 6 степени слободе емулира постојеће стандардизоване роботе са 5 степени слободе и специјализоване роботе за обраду са 5 степени слободе, односно вертикалне и хоризонталне машине алатке код којих оријентацију алата остварују осе ротација главног вретена. Усвојени концепт елементарне роботске ћелије за обраду је омогућио да се на систематизован начин моделирају кинематике робота вертикалне зглобне конфигурације са 5 и 6 степени слободе који емулирају различите машине алатке.

Према усвојеном концепту реконфигурабилних роботских ћелија за обраду, развијен је генерализован приступ кинематичког моделирања, који је представљао основ развоју реконфигурабилног система управљања са могућношћу програмирања у G -кôду. Поред могућности емулације различитих машина алатки, генерализовани приступ кинематичког моделирања омогућава динамичко реконфигурисање 6-осног робота, односно емулирање и хоризонталне и вертикалне 5-осне машине алатке у току једног задатка који захтева вишестрану обраду врло сложених делова у једном базирању. Овим приступом се избегавају и проблеми сингуларитета робота, као и проблеми ограничења кретања у зглобовима.

С обзиром на чињеницу да реконфигурабилни управљачки систем својом реконфигурабилношћу треба да покрије различите конфигурације роботских ћелија за обраду, али и да омогући интеграторима система да сами креирају нове конфигурације, поштујући принципе отворености, модуларности и скалабилности управљачке архитектуре, у овој дисертацији су постављена су два метода пројектовања реконфигурабилних управљачких система:

- Метод на бази примене готових хардверских и софтверских система отворене архитектуре;
- Метод на бази примене софтверских алата и хардверских технологија отворене архитектуре уз развој алгоритама управљања.

На бази постављене методологије, применом готових хардверских и

софтверских система отворене архитектуре развијен је и комплетно имплементиран реконфигурабилни управљачки систем на бази расположивог софтвера отворене архитектуре *EMC2*, *Ubuntu Linux* оперативног система са *real-time* језгром и *PC* хардвера, као готових решења. Према усвојеном концепту реконфигурабилне роботске ћелије за обраду, а на основу кинематичких модела добијених генерализованим приступом кинематичког моделирања, формирано је неколико конфигурација управљачког система. Посредством конфигурационог интерфејса, који је интегрални део *EMC2* софтвера, омогућен је избор одговарајуће конфигурације управљачког система. Током рада на овој дисертацији, посебна пажња је посвећена истраживању у области виртуелних обрадних система на бази робота и развоју симулационог окружења, као неодвојивог дела реконфигурабилног управљачког система роботских ћелија за обраду. Као резултат ових истраживања, добијен је софтверски систем који омогућава моделирање и анимацију виртуелних робота програмима обраде, односно сигнаlima са сензора позиција робота. Према иницијално усвојеном концепту реконфигурабилне роботске обрадне ћелије, конфигурисано је неколико виртуелних обрадних ћелија на бази робота, које су примењене за симулацију програма обраде, генерисаних посредством *CAM*-а за различите емулиране машине алатке.

Експерименти на реконфигурабилном управљачком систему, развијеном на бази примене готових хардверских и софтверских система отворене архитектуре, обрадом тест радних предмета сложене геометрије, функционалних и естетских површина, потврдили су да је систем директно применљив од стране оператера који имају искуства у *CNC* технологији и програмирању у *G*-коду, што чини један од најважнијих резултата ове дисертације. На примеру вишестране обраде су показане могућности реконфигурирања управљачког система, без физичких интервенција на роботу и хардверу управљачког система. Радни предмет је том приликом обрађен у само једном базирању, док је роботска ћелија емулирала вертикалну и хоризонталну глодалицу током обраде, при чему је одговарајућа конфигурација бирана посредством конфигурационог

интерфејса. Током развоја и експерименталне верификације потврђена је предност примене готових софтверских решења у реализацији управљачког система за обрадне ћелије на бази робота, која се огледа у могућности прецизне предикције перформанси и времена реализације управљања у раним фазама развоја. Са друге стране, уочен је недостатак оваквог приступа који се огледа у ограниченим могућностима примене одговарајућих сензора у циљу повећања тачности обраде и побољшања безбедности оператора током рада на роботској ћелији за обраду.

Према постављеној методологији развоја управљачког система, која подразумева примену софтверских алата и хардверских технологија отворене архитектуре уз развој алгоритама управљања, у дисертацији је дефинисан платформски независан концепт реконфигурабилног управљачког система са ослоном на већ развијене алгоритме и функције управљања у Лола институту. Развијени су одговарајући управљачки алгоритми који су постали основ за реализацију софтверских објеката и функција. Потреба за раздвајањем функција програмирања и симулације од функција управљања роботском ћелијом, како у софтверском тако и у хардверском смислу, одредила је приступ развоју система управљања и програмирања који се састоји из *real-time* софтверског подсистема који је задужен за управљање роботском ћелијом и *off-line* подсистема за писање и едитовање програма, проверу синтаксе и семантике кода, и симулацију програма обраде.

Захтев за динамичком реконфигурабилношћу управљачког система, као и захтев за имплементацијом софтверских функција за *on-line* компензацију грешака током обраде, уз коришћење додатних сензорских система условио примену *OROCOS* софтверских алата за креирање и повезивање софтверских компонената. Пратећи модуларну, хијерархијски организовану структуру постављеног концепта управљања, применом *OROCOS*-а су развијене основне софтверске компоненте, које посредством *CORBA*-е и *DCOM*-а могу бити статички или динамички додате или уклоњене из управљачког система. Примена постављеног морфолошког метода конфигурисања/реконфигурисања система је омогућена коришћењем *XML* конфигурационих

датотека, у којима корисник дефинише конфигурацију пре покретања система или је мења у току рада, без потребе за поновном компилацијом софтвера.

Виртуелно окружење за симулацију програма обраде и праћење рада роботске ћелије, развијено применом *OpenGL* библиотека у *Qt* платформски независном окружењу, један је од важнијих резултата ове дисертације. Развијена софтверска апликација омогућава моделирање и анимацију виртуелних робота програмима обраде, односно сигнаlima са сензора позиција робота.

Применом успостављене методологије, уз коришћење имплементираних основних компонената управљања, конфигурисан је први прототип експерименталног управљачког система. У циљу спровођења верификације функција и перформанси прототипа, са идејом да се избегну евентуални проблеми који могу настати услед неправилног функционисања софтвера и/или хардвера, верификација се одвијала у две фазе. Првом фазом су обухваћена иницијална тестирања на тест лабораторијском систему са 3 аналогна серво погона. Током ових тестирања потврђено је правилно функционисање прве три серво осе робота, које доминантно учествују у остваривање задате позиције, имају снажне погоне и које у каснијим фазама верификације на роботу могу утицати на безбедност. У другој фази верификације, експериментални прототип управљачког система је тестиран на роботу Лола 15, извршавањем сложених програмираних вишеосних кретања. Испитивања, која су вршена на тест лабораториском систему са 3 аналогна серво погона и роботу Лола 15, потврдила су правилно функционисање и временски детерминизам у извршавању развијених софтверских компонената управљања, као и могућност паралелног рада виртуелне роботске на одабраној хардверској платформи и оперативном систему.

Постављени методи, алгоритми, софтвер и развијени и тестирани експериментални прототипови управљачких система представљају солидну основу за даље правце истраживања у области вишеосне обраде роботима који ће се односити на имплементацију STEP-NC стандарда у управљачки

ЗАКЉУЧАК

систем, као развој и имплементацију *on-line* метода компензације статичких грешака обраде у управљачки систем, на бази интелигентних алгоритама модификације путање мерењем сила резања.

8 ЛИТЕРАТУРА

- [1] Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Usloy, G., Van Brassel, Reconfigurable Machining Systems, CIRP Annals, Vol.48/2/1999, pp. 527-540
- [2] Altintas, Y., Brecher, M., Weck, M., Witt, S., Virtual machine tool, CIRP Annals, Vol.54/2/2005, pp. 651-674.
- [3] Moriwaki, T. (2008) „Multi-functional machine tool“, CIRP Annals - Manufacturing Technology 57, 736-749.
- [4] Vergeest JSM, Tangelder JWH (1996) Robot machines rapid prototype. The industrial robot 23(5):17-20. DOI:10.1108/01439919610130328.
- [5] Shirase K, Tanabe N, Hirao M, Yasui T (1996) Articulated robot application in end milling of sculptured surface. JSME International Journal Series C 39(2):308-316.
- [6] Chen YH, Hu YN (1999) Implementation of a robot system for sculptured surface cutting. Part 1. Rough machining. Int J Adv Manuf Technol 15:624-629. DOI: 10.1007/s001700050111.
- [7] Hu YN, Chen YH (1999) Implementation of a robot system for sculptured surface cutting. Part 2. Finish machining. Int J Adv Manuf Technol 15:630-639. DOI: 10.1007/s001700050112.
- [8] Song Y, Chen YH (1999) Feature-based robot machining for rapid prototyping. Proc Instn Mech Engrs Part B 213(5):451-459. DOI:10.1243/0954405991516921.
- [9] Gerke W (2004) Milling robot with 3D vision system for styrofoam modelling. Proc IEEE International Conference on Mechatronics and Robotics part II, Aachen, pp 192-196.
- [10] Abele E, Kulok M, Weigold M (2005) Analysis of a machining industrial robot. Proc 10th International Scientific Conference on Production Engineering-CIM2005, Lumbarda, Croatia, pp II 1-11.
- [11] Abele E., Weigold M., Rothenbucher S. (2007) „Modeling and identification of an industrial robot for machining applications“, Annals of the CIRP 56(1):387-390.

ЛИТЕРАТУРА

- [12] Shin-ichi M, Kazunori S, Nobuyuki Y, Yoshinari O (1999) High-speed end milling of an articulated robot and its characteristics. *Journal of Materials Processing Technology*, 95:83-89. DOI:10.1016/S0924-0136(99)00315-5.
- [13] URL: <http://www.eew-protec.de/>.
- [14] Zhang, H.; Wang, J.; Zhang, G.; Gan, Z.; Pan, Z.; Cui, H. & Zhu, Z. (2005). Machining with flexible manipulator: toward improving robotic machining performance, *Proceedings of IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 1127-1132, ISBN 0-7803-9047-4.
- [15] Pan Z., Zhang H. (2008) „Robotics machining from programming to process control: a complete solution by force control“, *Ind Robot Int J* 35(5):400–409.
- [16] Olabi A, Bearee R, Gibaru O, Damak M (2010) Feedrate planning for machining with industrial six-axis robots. *Control Engineering Practice*. Article in press. DOI:10.1016/j.conengprac.2010.01.004.
- [17] Li W., Red E., Jensen G., Evans M. (2007) Reconfigurable mechanisms for application control (RMAC): applications. *Computer-Aided Design & Applications* 4(1-4):549-556. ISSN 1686-4360.
- [18] DePree J., Gesswein C. (2008) Robotic machining white paper project-Halcyon Development. URL: <http://www.halcyondevelop.com>.
- [19] Huang, H.-K., Lin, G.C.I. (2002) Development of a dual-robot system for prototype production. *International Journal of Production Research* 40(15):3751-3764. DOI:10.1080/00207540210146206.
- [20] Shirase K., Hirao M., Yasui T. (1994) „End Milling for Articulated Robot Application. 1st Report. Improvement of Machining Accuracy by Postprocess Error Compensation” *Transactions of the Japan Society of Mechanical Engineers Series C* 60(571):1070–1076, The Japan Society of Mechanical Engineers.
- [21] Shirase K., Hirao M., Yasui T. (1994) „End Milling for Articulated Robot Application. 2nd Report. Copy Milling Based on Sensor Feedback Teaching Program.” *Transactions of the Japan Society of Mechanical Engineers Series C* 60(571):1077–1083, The Japan Society of Mechanical Engineers.
- [22] Shirase K., Hirao M., Yasui T. (1995) „End Milling for Articulated Robot Application. 3rd Report, Improvement of Machining Accuracy by Robot

ЛИТЕРАТУРА

- Calibration” Transactions of the Japan Society of Mechanical Engineers Series C 61(581):259–265, The Japan Society of Mechanical Engineers.
- [23] Pan Z., Zhang H. (2008), „Robotic machining from programming to process control: a complete solution by force control.” *Industrial Robot: An International Journal*, 35(5):400–409.
- [24] Milutinovic D., Glavonjic M., Slavkovic N., Dimic Z., Zivanovic S., Kokotovic B., Tanovic Lj. (2011) Reconfigurable robotic machining system controlled and programmed in a machine tool manner. *International Journal of Advanced Manufacturing Technology* 53(9-12):1217-1229. DOI:10.1007/s00170-010-2888-8.
- [25] Milutinovic D, Glavonjic M, Zivanovic S, Dimic Z, Slavkovic N (2009) Development of robot based reconfigurable machining system. Proc 33rd Conference on Production Engineering of Serbia, Belgrade, pp 151-155 ISBN 978-86-7083-662-4.
- [26] Dimic Z, Milutinovic D, Zivanovic S, Kvrđic V (2015) „Virtual environment in control and programming system for reconfigurable machining robot”, *Tehnicka Gazette*, prihvaćeno za objavljivanje u Vol. 23/No. 6 (2016) pod brojem DOI: 10.17559/TV-20150210133556.
- [27] Slavkovic N., Milutinovic D., Glavonjic M. (2014) A method for off-line compensation of cutting force-induced errors in robotic machining by tool path modification. *International Journal of Advanced Manufacturing Technology* 70:2083–2096. DOI:10.1007/s00170-013-5421-z.
- [28] Славковић Н (2015) Идентификација, моделирање и компензација грешака услед статичке попустљивости робота за обраду, Докторска дисертација, Универзитет у Београду, Машински факултет, Београд.
- [29] Z. M. Bi & Sherman Y. T. Lang & M. Verner & P. Orban (2008) Development of reconfigurable machines. *International Journal of Advanced Manufacturing Technology* 39:1227–1251. DOI:10.1007/s00170-007-1288-1.
- [30] Glavonjic, M., Milutinovic, D., Zivanovic, S., Dimic, Z., Kvrđic V. (2010) Desktop 3-axis parallel kinematic milling machine. *The International Journal of Advanced Manufacturing Technology* Volume 46, Numbers 1-4 / January, 2010, str. 51-60, DOI: 10.1007/s00170-009-2070-3.

ЛИТЕРАТУРА

- [31] Glavonjić, M., Milutinović, D., Kvrđić, V., Dimić, Z., Živanović, S. (2006) Troosna vertikalna glodalica sa paralelnom kinematikom. Tehničko rešenje, Univerzitet u Beogradu, Mašinski fakultet.
- [32] Glavonjić, M., Milutinović, D., Kvrđić, V., Živanović, S., Dimić, Z., Slavković, N. (2012): Mini laboratorijska i edukaciona stona troosna glodalica sa paralelnom kinematikom. Tehničko rešenje, Univerzitet u Beogradu, Mašinski fakultet.
- [33] Milutinović, D., Glavonjić, M., Slavković, N., Dimić, Z., Kokotović, B., Živanović, S. (2010) Rekonfigurabilni obradni sistem na bazi robota za višeosnu obradu delova većih gabarita sa složenim estetskim i funkcionalnim površinama od mekših materijala srednje i niže klase tačnosti. Tehničko rešenje, Univerzitet u Beogradu, Mašinski fakultet.
- [34] V. Kvrđić, Z. Dimić, V. Cvijanović, D. Ilić, M. Bućan (2012) A Control Algorithm for a Vertical 5-Axis Turning Centre. The International Journal of Advanced Manufacturing Technology, Volume 61, Numbers 5-8, pp. 569-584. DOI: 10.1007/s00170-011-3737-0.
- [35] Kvrđić, V., Dimić, Z., Cvijanović, V., Vidaković, J., Kablar, N. (2014) A control algorithm for improving the accuracy of five-axis machine tools. International Journal of Production Research. DOI: 10.1080/00207543.2013.858194.
- [36] Kvrđić, V., Dimić, Z., Milićević, M., Cvijanović, V., Ilić, D. (2009) "Upravljačka jedinica otvorene arhitekture za upravljanje mašinama alatkama i robotima. Tehničko rešenje, Lola institut, Beograd.
- [37] Ferenc, G., Dimić, Z., Lutovac, M., Vidaković, J., Kvrđić, V. (2013) Open Architecture Platforms for the Control of Robotic Systems and a Proposed Reference Architecture Model. Transactions of Famena vol. 37 br. 1, pp. 89-100. ISSN 1333-1124.
- [38] Ferenc, G., Dimić, Z., Kvrđić, V., Cvijanović (2014) Sistem za upravljanje industrijskim robotima realizovan primenom savremenih softverskih alata za rad u realnom vremenu. Tehničko rešenje, Lola institut, Beograd.
- [39] Koren, Y., Shpitalni, M. (2010). „Design of reconfigurable manufacturing system”, Journal of Manufacturing Systems 29, 130-141.
- [40] Koren Y., Kota S., „Reconfigurable Machine Tool”, US Patent 5 943 750, 1997.

ЛИТЕРАТУРА

- [41] Katz, R., Yook, J., Koren, Y.(2010). „Control of Non-Orthogonal Reconfigurable Machine Tool“, ACME Journal of Dynamics, Measurement, and Control, Vol.126, 397-405.
- [42] Brecher, C., Verl, A., Lechler, A., & Servos, M. (2010), „Open control systems: state of the art“, German Academic Society for Production Engineering, 247–254.
- [43] Brogardh, T. (2007) „Present and future robot control development—An industrial perspective“, Annual Reviews in Control 31, 69–79.
- [44] Gu, J. S., & de Silva, C. W. (2004) „Development and implementation of a real-time open-architecture control system for industrial robot systems“, Engineering Applications of Artificial Intelligence 17, 469–483.
- [45] Landers, R., Min, B., & Koren, Y. (2001) „Reconfigurable Machine Tools“, CIRP Annals Manufacturing Technology, 269-274.
- [46] Pritschow, G., Altintas, Y., Jovane, F., Koren, Y., Mitsuishi, M., Takata, S., et al. (2001) „Open Controller Architecture - Past, Present and Future“, Annals of the CIRP, 463 - 470.
- [47] Oldknow, K. D., & Yellowley, I. (2001) „Design, implementation and validation of a system for the dynamic reconfiguration of open architecture machine tool controls“, International Journal of Machine Tools & Manufacture 41, 795–808.
- [48] Teltz, R. W., & Elbestawi, M. A. (1997) „Design basis and implementation of an open architecture machine tool controller“, Transactions of NAMRI/SME XXV, 229–304.
- [49] Li W., Red E., Jensen G., Evans M. (2007) „Reconfigurable mechanisms for application control (RMAC): applications“, Comp Aided Des Appl 4(1–4):549–556.
- [50] International Federation of Robotics.
URL: <http://www.ifr.org/industrial-robots/statistics>.
- [51] Thomessen T.; Sanches P. K., Lien T. K. (2004). Intuitive Robot Programming, Proceedings of 35th International Symposium on Robotics, ISBN 0-7695-0751-6.

ЛИТЕРАТУРА

- [52] Webb G., Morel M. K., Robots: The Lower Cost, More Flexible Process Improvement Alternative to CNC Machine Tools,
URL: <http://www.Robotmachining.com>.
- [53] URL: <http://www.staubli.com>.
- [54] URL: <http://www.kuka.com>.
- [55] URL: <http://new.abb.com/products/robotics>.
- [56] URL: <http://www.fanuc.eu/bg/en/robots>
- [57] Solvang B., Sziebig G., Korondi P (2008) „Robot Programming in Machining Operations” из књиге „Robot Manipulators”, едитор: Marco Ceccarelli, ISBN 978-953-7619-06-0, издавач: InTech.
- [58] URL: <http://www.robotics.org/robotic-content.cfm/Robotics/Halcyon-Development-RIA/id/43>.
- [59] URL: http://www.kuka.com/nl_media/06/artikel_camrob_en.pdf.
- [60] URL: <http://robot.fanucamerica.com/products/vision-software/ROBOGUIDE-simulation-software.aspx>.
- [61] URL: <http://www.motoman.com/datasheets/G-CodeEG.pdf>.
- [62] URL: <http://new.abb.com/products/robotics/application-software/machining/robotware-machining-fc>.
- [63] URL: <http://www.robotmaster.com>.
- [64] URL: <http://www.delcam.com/software/robotics/overview/index.asp>.
- [65] Li W., Red W.E., Jensen C.G., Evans M. (2007) „Reconfigurable Mechanisms for Application Control (RMAC): Architecture.” Computer Aided Design and Applications, 4(1-4):539–548.
- [66] Li W., Red W.E., Jensen C.G., Evans M. (2007) „Reconfigurable Mechanisms for Application Control (RMAC): Applications.” Computer Aided Design and Applications, 4(1-4):549-556.
- [67] SEVENTH FRAMEWORK PROGRAMME: Plug-and-produce **CO**mponents and **ME**thods for adaptive control of industrial robots enabling cost effective, high precision manufacturing in factories of the future.
URL: <http://www.cometproject.eu>.

ЛИТЕРАТУРА

- [68] Lee R.S., She C.H. (1997) Developing a postprocessor for three types of five-axis machine tools, *International Journal of Advanced Manufacturing Technology*, 13 /9 658-665.
- [69] ISO 841:2001 Industrial automation systems and integration -- Numerical control of machines -- Coordinate system and motion nomenclature.
- [70] A. Affuard, E. Duc, C. Lartigue, J. M. Langeron, P. Bourdet (2004) Avoiding five-axis singularities using tool path deformation, *International Journal of Machine Tools and Manufacture*, 44(4):415–425. DOI: 10.1016/j.ijmachtools.2003.10.008.
- [71] Craig JJ (1989) „Introduction to robotics: mechanics and control“, 2nd ed. Addison- Wesley ISBN 0-201-09528-9.
- [72] Paul RP (1981) „Robot Manipulators: mathematics, programming and control“, The MIT Press ISBN 0-262-16082-X.
- [73] Fu KS, Gonzalez RC, Lee CSG (1987) „Robotics: control, sensing, vision, and intelligence“.
- [74] Lee R.S., She C.H. (1997) „Developing a postprocessor for three types of five-axis machine tools.“, *International Journal of Advanced Manufacturing Technology* 13(9):658-665.
- [75] R. Manseur, K. Doty (1992) Fast inverse kinematics of five-revolute-axis robot manipulators, *Mechanism and Machine Theory*, 27(5):587–597. DOI: 10.1016/0094-114X(92)90047-L.
- [76] A. Pashkevich (1997) Real-time inverse kinematics for robots with offset and reduced wrist, *Control Engineering Practice*, 5(10):1443–1450. DOI: 10.1016/S0967-0661(97)00142-1.
- [77] C. Innocenti, C. Parenti (1994) Exhaustive enumeration of fully parallel kinematic chains, *Dynamic System and Control*, 55(2):1135–1141.
- [78] URL: <http://www.linuxcnc.org>.
- [79] OROCOS - **O**pen **R**obot **C**ontrol **S**oftware. URL: <http://www.orocos.org>.
- [80] URL: <http://www.isd.mel.nist.gov/projects/rcslib>.
- [81] OMG, Unified Modeling Language™ (UML®),
URL: <http://www.omg.org/spec/UML>.

ЛИТЕРАТУРА

- [82] Kvrđić, V. (1999) Razvoj inteligentnog sistema za upravljanje i programiranje industrijskih robota, Doktorska disertacija, Univerzitet u Beogradu, Mašinski fakultet, Beograd.
- [83] Pavlović, M. (1994) Viši programski jezik za multirobotske operacije, Magistarski rad, Univerzitet u Beogradu, Elektrotehnički fakultet.
- [84] Alfred Aho, Ravi Sethi, Jeffrey Ullman (1986) Compilers Principle, Techniques and Tools. Addison Wesley.
- [85] Hansen, Per Brinch (1985) Brinch Hansen on Pascal Compilers. New Jersey: Prentice-Hall Inc.
- [86] Suk-Hwan Suh, Seong-Kyoon Kang, Dae-Hyuk Chung, Ian Stroud (2008). Interpolator. In: Theory and Design of CNC Systems, 69–106. London: Springer-Verlag London Limited.
- [87] STEPTOOLS. URL: <http://www.steptools.com>.
- [88] Toolkit for STEP-NC. URL: <https://github.com/ArcEye/iso-14649-toolkit>.
- [89] Zubizarreta A, Portillo E, Cabanes I, Orive D, Marcos M, Pinto Ch. (2013) Real-time environment design for testing advanced control approaches in parallel robots. Int J Adv Manuf Technol (2013) 67:1459–1475. DOI 10.1007/s00170-012-4581-6.
- [90] PTC Creo. URL: <http://www.ptc.com>.

БИОГРАФИЈА

Зоран Димић, дипл. инж. ел. је рођен 8. октобра 1968. године у Београду, где и данас живи са супругом Драганом, сином Николом и ћерком Јелисаветом. Основну школу, у којој је показао прва интересовања за електронику, завршио је са одличним. Средњу електротехничку школу "Никола Тесла" је завршио 1987. године са одличним успехом, после чега одлази у школу резервних официра на одслужење војног рока. По повратку, 1988. године отпочиње студије на Електротехничком факултету.

Октобра месеца 1991. године бива мобилисан у тадашњу ЈНА. Јесен и зиму исте године проводи на простору источне Славоније, где учествује у ратним дејствима. По демобилизацији, кратко време ради у модној агенцији „Гламур“ као технички сарадник, а након тога га ангажује фирма „ENERGOREMONT CO.“ у којој је до октобра 1988. године радио на пословима монтаже, конфигурисања и одржавања управљачких система за термоенергетска постројења.

Започете студије електротехнике наставља октобра 1988. године и након апсолвентуре, на позив Др Валдимира Квргића, тадашњег директора „Лола фабрике робота“, маја 2000. године се запошљава у "Лола корпорацији", на позицији инжењера за развој системског софтвера робот контролера и система за *off-line* програмирање робота. Дипломирао је на смеру Електроника, Електротехничког факултета Универзитета у Београду септембра .2001 године. Дипломски рад је урадио из предмета Импулсна и дигитална електроника на тему „Одређивање угловних позиција оса робота“ под менторством проф. др Дејана Живковића. Од августа 2007. године до данас је запослен у „Лола институту д.о.о.“ као истраживач и руководилац програма развоја управљачких система за роботе и машине алатке.

Током истраживачког рада је учествовао у већем броју научних и стручних пројеката из области индустријских робота и машина алатки, који су се односили на системе управљања и програмирања отворене архитектуре за роботе и машине алатке. Самостално или као коаутор је објавио већи број

радова на домаћим и међународним скуповима и у домаћим и међународним часописима. Такође је коаутор и већем броју техничких решења.

За постигнуте научне и стручне резултате је два пута, 2005. и 2010. године, награђиван Годишњим наградама Привредне коморе Београда за техничка решења и унапређења, а 2006. године је награђен Наградом града Београда за област наука-проналазаштво. Као члан истраживачког тима, 2011. године је награђен Наградом за најбољу технолошку иновацију за освојено четврто место у укупном пласману.

Прилог 1.

Изјава о ауторству

Потписани-а Зоран Ж. Димић

број уписа D7/08

Изјављујем

да је докторска дисертација под насловом

УПРАВЉАЧКИ СИСТЕМ ОТВОРЕНЕ АРХИТЕКТУРЕ

РЕКОНФИГУРАБИЛНИХ РОБОТСКИХ ЋЕЛИЈА ЗА ОБРАДУ

- резултат сопственог истраживачког рада,
- да предложена дисертација у целини ни у деловима није била предложена за добијање било које дипломе према студијским програмима других високошколских установа,
- да су резултати коректно наведени и
- да нисам кршио/ла ауторска права и користио интелектуалну својину других лица.

Потпис докторанда

У Београду, 04.04.2016. године

Прилог 2.

Изјава о истоветности штампане и електронске верзије докторског рада

Име и презиме аутора	<u>Зоран Ж. Димић</u>
Број уписа	<u>D7/08</u>
Студијски програм	<u>Машинско инжењерство</u>
Наслов рада	<u>УПРАВЉАЧКИ СИСТЕМ ОТВОРЕНЕ АРХИТЕКТУРЕ РЕКОНФИГУРАБИЛНИХ РОБОТСКИХ ЋЕЛИЈА ЗА ОБРАДУ</u>
Ментор	<u>Др Драган Милутиновић, редовни професор, Универзитет у Београду, Машински факултет</u>

Потписани Зоран Ж. Димић

изјављујем да је штампана верзија мог докторског рада истоветна електронској верзији коју сам предао/ла за објављивање на порталу **Дигиталног репозиторијума Универзитета у Београду**.

Дозвољавам да се објаве моји лични подаци везани за добијање академског звања доктора наука, као што су име и презиме, година и место рођења и датум одбране рада.

Ови лични подаци могу се објавити на мрежним страницама дигиталне библиотеке, у електронском каталогу и у публикацијама Универзитета у Београду.

Потпис докторанда

У Београду, 04.04.2016. године

Прилог 3.

Изјава о коришћењу

Овлашћујем Универзитетску библиотеку „Светозар Марковић“ да у Дигитални репозиторијум Универзитета у Београду унесе моју докторску дисертацију под насловом:

УПРАВЉАЧКИ СИСТЕМ ОТВОРЕНЕ АРХИТЕКТУРЕ
РЕКОНФИГУРАБИЛНИХ РОБОТСКИХ ЋЕЛИЈА ЗА ОБРАДУ

која је моје ауторско дело.

Дисертацију са свим прилозима предао/ла сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигитални репозиторијум Универзитета у Београду могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons) за коју сам се одлучио/ла.

1. Ауторство
2. Ауторство - некомерцијално
3. Ауторство – некомерцијално – без прераде
4. Ауторство – некомерцијално – делити под истим условима
5. Ауторство – без прераде
6. Ауторство – делити под истим условима

(Молимо да заокружите само једну од шест понуђених лиценци, кратак опис лиценци дат је на полеђини листа).

Потпис докторанда

У Београду, 04.04.2016. године
