

Beroepsproduct Interface: Robotarmbesturing

Bram Knippenberg, Bas Lijnse, Jorg Visch
11 februari 2020

1. INLEIDING

In deze opdracht ga je een software interface (API) ontwerpen en bouwen voor de besturing van een LynxMotion robotarm vanuit een C++-programma in het Robot Operating System (ROS). Deze opdracht maak je alleen of in een tweetal.

De LynxMotion AL5D is een eenvoudige robotarm met zes vrijheidsgraden zonder feedback. De robotarm gebruikt een generieke servocontroller SSC-32U voor het aansturen van zijn servo's. Er kan met deze controller worden gecommuniceerd via een eenvoudig protocol over een seriële poort. Dit protocol biedt enkele low-level functionaliteiten. Voor het uitvoeren van complexere taken is er een high-level interface benodigd.

In dit document vind je de opdrachtomschrijving, de gestelde eisen met bijbehorende normering en enige informatie met betrekking tot het inleveren van de opdracht. Zie appendix A voor technische details betreffende de robotarm.

Met betrekking tot het inleveren gelden de volgende eisen:

- Inleveren via iSAS voor de deadline (zie toetsrooster iSAS voor de exacte datum en tijdstip).
- Alle onderdelen dienen gebundeld in één ZIP- of RAR-bestand te worden aangeleverd. (let op, iSAS gaat niet goed om met tar.gz, dus altijd in zip of rar)
- De source code moet in de vorm van een git-repository (clone) worden aangeleverd.
- Documenten worden in PDF-formaat aangeleverd.
- Losse plaatjes zijn alleen toegestaan indien er naar verwezen wordt in de documentatie. Afbeeldingen alleen in PNG-formaat.

Iedere groep moet een demonstratie geven van de gerealiseerde interface, waarbij de uitwerking mondeling wordt toegelicht.

2. OPDRACHTBESCHRIJVING

Voor diverse projecten is het wenselijk te beschikken over een robotarm. De LynxMotion AL5D is gekozen als eerste arm die hiervoor ingezet gaat worden. De robotarm moet ingezet kunnen worden in systemen met verschillende subsysteem-, sensor- en actuatorconfiguraties. Voor alle systemen geldt echter dezelfde basisopzet met onderstaande twee subsystemen:

- Motion-planner
De motion-planner maakt op basis van sensor- en planningsinput beslissingen m.b.t. te ondernemen acties en laat deze door de robotarm uitvoeren.
- Robotarm
De robotarm voert de door de motion-planner aangegeven acties uit.

Deze opdracht richt zich op de volgende zaken:

- Het ontwerp en de implementatie van een (C++)-interface die een aantal basisfuncties van de robotarm ontsluit, zodat deze door de motion-planner kan worden ingezet.
 - Let op: De *motion-planner* zelf is *geen* onderdeel van deze opdracht.
- Inzage in de bruikbaarheid van de gekozen robotarm. Om een idee te krijgen van de bruikbaarheid zijn er enkele eisen gesteld aan de performance en moet worden bepaald in hoeverre deze eisen haalbaar zijn.

De volgende paragrafen beschrijven de gestelde eisen aan de uitwerking, tevens is de puntentelling gegeven. Raadpleeg de docent (dat is de opdrachtgever) bij onduidelijkheden of onmogelijkheden bij de gestelde eisen en doe eventuele verbetervoorstellen.

2.1. BENODIGDE FUNCTIONALITEIT

De functionaliteit die door de interface moet worden geboden bestaat uit de onderstaande zaken. Eventuele eisen zijn opgenomen in de beschrijving.

2.1.1. SERVOAANSTURING

ID	Omschrijving	Prio	Punten
SA01	Elke vrijheidsgraad moet worden ontsloten, waarbij de bijbehorende servo's naar de opgegeven positie (hoek, in graden) kunnen worden verplaatst.	Must	4
SA02	De verplaatsing moet met verschillende snelheden uitgevoerd kunnen worden. Hierbij moet de opgegeven positie in de opgegeven tijd (in milliseconden) worden bereikt.	Should	4
SA03	Als de gripper van AL5D-robotarm naar een locatie (een set van samengestelde servohoeken ¹) wordt gestuurd moet deze binnen 2,3 seconden worden bereikt.	Should	3
SA04	De software moet op een eenvoudige manier rekening houden met verschillen in robotarmen wat betreft de draaiing. Bijv. bij de optie <i>straightup</i> (eis PO03) staat elke arm precies rechtop. De software houdt dus rekening met verschillen in standen van de servo's.	Must	2

2.1.2. VOORGEPROGRAMMEERDE POSITIES

De arm moet naar een aantal voorgeprogrammeerde posities kunnen worden gestuurd (zie appendix A):

ID	Omschrijving	Prio	Punten
PO01	Park: Een positie waarin de arm veilig kan worden uitgeschakeld.	Must	3
PO02	Ready: De "Ready"-stand kan als basispositie voor het uitvoeren van werk worden gebruikt.	Must	3
PO03	Straight up: Een positie waarin de robotarm (gecentreerd) volledig rechtop staat.	Must	3

VEILIGHEID

Om veilig met de arm te kunnen werken zijn er enkele veiligheidsfunctie gedefinieerd.

ID	Omschrijving	Prio	Punten
VE01	Noodstop: De arm wordt zo snel mogelijk gestopt, waarbij de huidige actie wordt afgebroken. Alle openstaande acties van de robotarm worden geannuleerd.	Must	5
VE02	Beperking range of motion: De software beperkt de beweging van de servo's tot een veilige range. Zie de tabel in appendix A voor de veilige ranges.	Must	3
VE03	Opstart-initialisatie: Bij het opstarten van de robotarm gaat deze gegarandeerd naar de Park-positie. De verplaatsingssnelheid is hierbij gelimiteerd tot de helft van de maximumsnelheid.	Should	3

¹rekening houdend met de onder veiligheid afgebakende range of motion

2.1.3. OPERATIONELE INFORMATIE

Het systeem ontsluit operationele informatie via *rosout* op verschillende log-niveaus.

ID	Omschrijving	Prio	Punten
INF01	De interface geeft toegang tot informatie over de operationele <i>toestand</i> van het arm-systeem via <i>rosout</i> op <i>INFO</i> niveau. De weergegeven toestanden moeten overeenkomen met de toestanden van de protocol state machine bij US02. De informatie wordt geboden in het volgende formaat " <i>STATE: {statename}</i> ".	Must	3
INF02	Geef <i>event</i> informatie via <i>rosout</i> op <i>DEBUG</i> niveau. Hierbij zijn in ieder geval de events opgenomen zoals zichtbaar zijn in de protocol state machine bij US02. De informatie wordt geboden in het volgende formaat " <i>EVENT: {eventname}</i> ".	Must	3
INF03	Geef een <i>WARNING</i> via <i>rosout</i> indien een opdracht aan de robotarm niet kan worden uitgevoerd binnen de gehaalde tijd ² en geef daarbij de wel verwachte tijd van afhandeling aan. De informatie wordt geboden in het volgende formaat " <i>QoS-Warning: {warning message}</i> ".	Could	2

2.1.4. EXTRA

ID	Omschrijving	Prio	Punten
EX01	Queuing van opdrachten: Er kunnen meerdere opdrachten worden toegevoegd aan een wachtlijst. De queue kan worden geleegd. Bij een noodstop wordt de queue geleegd.	Would	3
EX02	Eigen innovatie: Zelf in te vullen extra functionaliteiten.	Would	4

2.1.5. DOCUMENTATIE

Voor inzage in het ontwerp, mogelijke doorontwikkeling en beoordeling van inzetbaarheid moeten er een aantal zaken zijn gedocumenteerd. De documentatie moet in één document worden aangeleverd. In dit document worden de verschillende onderdelen nader toegelicht met een samenhangend verhaal op basis van de gevraagde UML diagrammen. Gemaakte keuzes worden bij de relevante onderdelen beargumenteed.

2.1.6. BESCHRIJVING VAN HET SYSTEEM

ID	Omschrijving	Prio	Punten
SY01	Use cases, inclusief decompositie in subsystemen.	Must	4
SY02	Beschrijving van het systeem, de gebruikte interfaces en poorten aan de hand van een componentdiagram.	Must	4
SY03	Een beschrijving van de interfaces op voldoende niveau voor gebruik in de gevraagde timingdiagrammen.	Should	3
SY04	Een beschrijving van het systeemgedrag met behulp van state diagram(s).	Should	4

2.1.7. INZICHT IN GEBRUIK

Voor het gebruik van de interface is de volgende informatie beschreven:

²Bijvoorbeeld gebaseerd op de resultaten van de timing-analyse voor QS02

ID	Omschrijving	Prio	Punten
US01	Samenwerking tussen de diverse deelsystemen. Op basis van een sequence diagram die de opstart-initialisatie laat zien.	Should	3
US02	Inzicht in resource beschikbaarheid in de vorm van stateafhankelijke beperkingen en geboden functies (events waar op gereageerd kan worden) op basis van een <i>Protocol State Machine</i> behorende bij de high level interface.	Must	4

2.1.8. INZICHT IN BRUIKBAARHEID

Voor de inzet van de robotarm is het van groot belang dat er inzicht wordt verschaft met betrekking tot de bruikbaarheid, uitgedrukt in termen van Quality of Service.

ID	Omschrijving	Prio	Punten
QS01	Weergave van de vereiste Quality of Service, vastgelegd op use case niveau.	Must	3
QS02	Beargumenteer of de vereiste Quality of Service ook daadwerkelijk wordt geboden ³ . Doe dit op basis van timing-diagrammen (en eventueel ondersteunende sequence-diagrammen), voor het scenario waarin de gripper van de robotarm naar een opgegeven positie wordt verplaatst. Ga er hierbij vanuit dat de diverse systeemonderdelen zijn geïnitieerd en de robotarm stilstaat op een locatie binnen de afgebakende range-of-motion.	Should	6

2.2. IMPLEMENTATIE

Voor de implementatie maak je gebruik van C++ op **ROS Melodic Morenia** met gebruik van de **Boost libraries** (bij ROS wordt al een versie van Boostversie geïnstalleerd, overschrijf deze *niet* met een andere versie). Eventueel gebruik van andere libraries moet altijd met de docent overlegd worden. De implementatie moet bestaan uit de volgende onderdelen:

ID	Omschrijving	Prio	Punten
IM01	De software biedt een <i>high-level</i> interface die de vereiste functionaliteit ontsluit via de ROS-communicatieinfrastructuur (topics, services, actions etc.).	Must	4
IM02	De software bevat een interne <i>low-level</i> driver (in de vorm van een Dynamically Linked Library) voor het direct ontsluiten van de bestaande functies van de robotarm. Dit is een 1-op-1 koppeling met de door de hardware geboden functies die benodigd zijn voor de implementatie van de gevraagde functionaliteit.	Must	4
IM03	De software zorgt voor encapsulatie van de geserialiseerde aansturing.	Should	3
IM04	De applicatie is voorzien van API-documentatie met behulp van doxygen.	Could	3
IM05	De applicatie is te bouwen met catkin op ROS Melodic Morenia.	Must	1
IM06	De applicatie wordt gebouwd met C++ volgens de Object Oriented principes die je geleerd hebt bij eerdere courses.	Must	1

³In het geval er niet aan de eisen kan worden voldaan moet inzichtelijk worden gemaakt welke Quality of Service er wel kan worden geboden.

2.3. DEMONSTRATIE

Om de functionaliteit aan te tonen en zonodig toe te lichten moet er een demo-applicatie geschreven worden met de volgende functionaliteit:

ID	Omschrijving	Prio	Punten
DE01	De demo gebruikt de interface om de robotarm het volgende scenario te laten doorlopen: <ul style="list-style-type: none">– Opstart en initialisatie.– Korte pauze.– Voorgeprogrammeerde posities: Ready en Straight up.– Verschillende opeenvolgende armverplaatsingen naar verschillende locaties.– Noodstop (werking moet duidelijk zichtbaar zijn).– Park.	Must	4
DE02	De demo-applicatie toont expliciet het gerealiseerd QoS aan.	Could	3

3. BEOORDELING

Met de in het voorgaande beschreven eisen zijn in totaal 100 punten te verdienen. De must-eisen leveren minder dan 55 punten op, je zult dus extra eisen moeten implementeren. De punten per eis die je scoort is afhankelijk van de documentatie en implementaties (m.a.w. heb je aan een eis voldaan, maar de documentatie of implementatie voldoet minder aan de eisen zoals we die tijdens de courses leren, krijg je niet het volledige puntenaantal).

Als je niet aan **alle must-eisen** voldoet krijg je maximaal 50 punten. Het cijfer is het aantal punten gedeeld door 10.

A. TECHNISCHE INFORMATIE AL5D ROBOTARM

- AL5D-arm: <http://www.lynxmotion.com/c-130-al5d.aspx>
- SSC-32U USB Servo Controller: <http://www.lynxmotion.com/p-1032-ssc-32u-usb-servo-controller.aspx>
- SSC-32U protocol: http://www.lynxmotion.com/images/data/lynxmotion_ssc-32u_usb_user_guide.pdf

A.1. RANGE OF MOTION

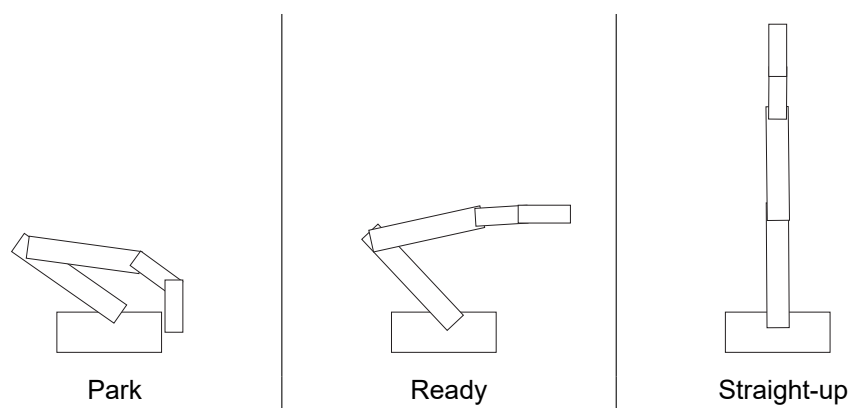
De onderstaande tabel toont van iedere servo het SSC-32 servo-connectienummer en de servo-posities. Deze zijn voorzien van naam en hoek (in graden). Tussengliggende hoeken zijn toegestaan, maar de overschrijding van de uiterste waarden niet.

Nr	Onderdeel	Positie	Hoek (deg)
0	Base	Left	-90
		Middle	0
		Right	90
1	Shoulder	Backwards	-30
		Vertical	0
		Horizontal	90
2	Elbow	Straight	0
		Sharp down	90
		Inwards	135
3	Wrist	Up	90
		Straight	0
		Down	-90
4	Gripper	Fully open	-
		Fully closed	-
5	Wrist rotate	Left	-90
		Middle	0
		Right	90

A.2. POSITIES

Figuur A.1 toont de vooraf bepaalde basisposities.

- De robotarm moet bij aan- en uitschakelen met lage snelheid in de Park-positie worden geplaatst.
- De Ready stand kan als basispositie voor het uitvoeren van werk worden gebruikt.



Figuur A.1.: De basisposities van de AL5D arm