Doc ID 18102902R04

Doc Creation Date 19 APR 2018

Doc Revision

Doc Revision Date

Doc Status Final

Work package	Deliverable ID
WP4, Control layer design and development	**D4.2, Motion control requirements and specification (final iteration)**

Summary

This deliverable provides a summary of the final requirements for motion control applications utilizing centralized and decentralized control solutions.
The report follows directly Deliverable 4.1 (Motion control requirements and specification (first iteration)) according to the iterative process described in Task 2.3.
The requirements are strictly related to the needs of pilot plants, demonstrators and use cases by specifying them in detail with respect to control technologies and design tools. They will be used to test the results control related building blocks.

(Main)Authors UNIBS Team (Antonio Visioli)

Keywords
Control, Drives, Dynamics, Vibrations, Residual Oscillations, Bandwidth, Iterative Learning Control, Repetitive Control, Robustness, Trajectory Planning, Observers, Filters, Vision

Coordinator Sioux CCM
Tel. 0031 (0)40.263.5000
E-Mail info@I-MECH.eu
Internet www.I-MECH.eu

Table of contents

(Open) Issues & Actions

Open Issues (and related actions) that need central attention shall be part of a file called "IAL - Issues & Action List – Partners" which is can be found in the Google Drive Partner Zone.

ID	Description	Due Date	Owner	IAL ID

Document Revision History

Revision	Status	Date	Deliverable leader	Description of changes	IAL ID / Review ID
R01	Draft	14-05-18	Visioli	Initiate	
R02	Draft	08-08-18	Visioli	Various contributions added	
R03	Draft	28-09-2018	Visioli	Various contributions and Section 2 added	
R04	Draft	18-10-2018	Visioli	Section 4 added	
R05	Final	30-10-2018	Visioli	New Section 4 added	

Contributors

Revision	Acronym of Partner	Contributor	Description of work
R02	NXP	Van der Veen	Pilot 2 requirements
	IMA	Tagliapietra	Pilot 3 requirements
	PHI	Smeets	Pilot 4 requirements

	ECS	Ribeiro	Demonstrator 2 requirements
	GEF	Colombo	Use case 1.1 requirements
	FAG	Arenas	Use case 1.2 requirements
	TECO	Goubej	Use case 1.3 requirements
	GMV	Estremera	Use case 2.1 requirements
	ZAPUNI	Goubej	Use case 2.2 requirements
	SCC	Pulles	Pilot 1 requirements
	VIS	Hickey	Demonstrator 1 requirements

Document control

		Status							
		Revision							
Reviewer Name	Role	Selection							

File Locations (cross reference to I-MECH documents)

Via URL with a name that is equal to the document ID, you shall introduce a link to the location (either in Partner Zone or CIRCABC)

URL	Filename	Date

Abbreviations & Definitions

Abbreviation	Description
SISO	Single-input-single-output
MIMO	Multiple-input-multiple-output
PI	Proportional-integral
PID	Proportional-integral-derivative

RC	Repetitive control
ILC	Iterative learning control
IP	Internet Protocol
PLC	Programmable logic controller

Definition	Description

1 About this document

This deliverable is related to the Task 4.1 and describes the final control system requirements and specifications.
It follows directly Deliverable 4.1 "D4.1, Motion control requirements and specification (first iteration)", which has focused on the state-of-the-art of control algorithms applied to mechatronic systems and the future requirements of control systems, with a special emphasis in the pilot plant applications of the I-MECH project.
In this deliverable, the control requirements for pilot plants, demonstrators and use cases are given with much more detail and fixed. Further, technologies, languages and tools that will be used to satisfy the requirements are specified, so that the BB owners can take them into account in the development of the BBs. This is done by taking into account the expected improvements in the performance in order to achieve one of the main goals of the ECSEL JU, that is, the improvement of systems engineering in general and of electronics components employed for industrial production.
As for D4.1, this deliverable is strictly related to the activity of WP4 "Control Layer design and development", which aims at developing centralized and decentralized motion control strategies for mechatronic systems.
Then, D4.2 should be used, according to the procedures given in D6.2, to validate the performance of the building blocks directly related to the control layer, that is, BB6 "Self-commissioning velocity and position control loops", BB7 "Vibration control module", BB8 "Robust model-based multivariable control" and BB9 "Iterative and repetitive control module" and of those building blocks (BB10 "Control Specific Multi-many core Platform" and BB11 "RTOS for multi-many core platform") which are related to the HW/SW platforms which allows the implementation of the advanced control algorithms.

2 Linkage to other I-MECH deliverables

This deliverable is part of a set of deliverables that provide the overall requirements for the I-MECH project. In particular, Figure 1, borrowed from the WP2 highlights presentation at the review meeting in Brussels, shows the relationship between the deliverables related to the general requirements and the deliverables related to the specific work packages. The arrows specify the main linkages between the deliverables, but there are in any case connections that are not highlighted.
This deliverable receives inputs from D4.1 and D2.4 and it is developed in parallel with D3.2 and D5.2. In this context, final control requirements for pilots, demonstrators and use cases are outlined by taking into account that their intersection should constitute the base for the development of the control related building blocks (BB7-BB11, but also BB6, which is obviously related to the control layer as well). Then, the requirements should be satisfied by developing BBs according to the architectural and functional requirements for the I-MECH platform given in D2.4.
Together with D3.2 and D5.2, D4.2 serves as a reference for the validation and verification of the BBs, which should be performed by considering the guidelines detailed in D6.2.
Finally, it has to be noted that the description of the pilots and of their requirements, including the control ones, is specifically considered in D7.1. They are repeated here in order to have this deliverable more self-contained.

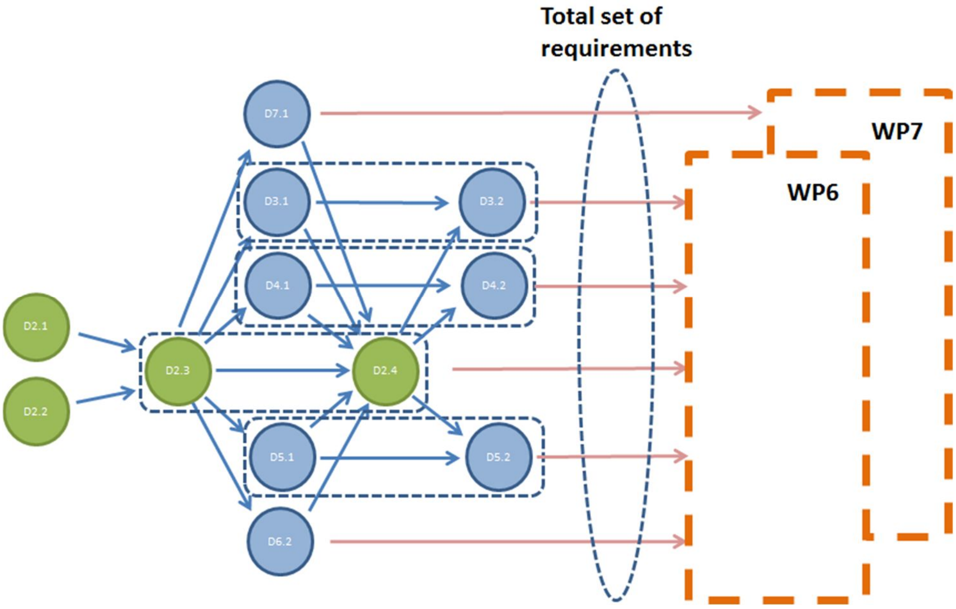

Figure 1. Relationship between the deliverables related to requirements.

3 Final control requirements

In order to summarize the final control requirements for the different pilots, demonstrators and use cases, and in order to clarify their relationships with the I-MECH Tasks and Building Blocks, the structured approach already followed in D4.1 (and in the other deliverables related to requirements, starting from D2.3) is used in the following table.

The employed code legend is as follows (each requirement ID is prefixed with rq- (for requirement), the deliverable ID, in this case D4.2, and the kind of requirement):

- rq-D4.2-P: Performance requirements
- rq-D4.2-T: Technical requirements
- rq-D4.2-R: Realization requirements

Code	Description
rq-D4.2-P.01	Stability
rq-D4.2-P.02	Overshoot
rq-D4.2-P.03	Set point tracking
rq-D4.2-P.04	Disturbance rejection
rq-D4.2-P.05	Noise attenuation
rq-D4.2-P.06	Bandwidth and sampling frequency
rq-D4.2-T.01	Signal filtering
rq-D4.2-T.02	Friction and inertia estimation

rq-D4.2-T.03	Feedforward compensation
rq-D4.2-T.04	Periodic harmonic disturbance compensation
rq-D4.2-T.05	Repetitive disturbance compensation
rq-D4.2-T.06	Oscillation compensation
rq-D4.2-T.07	Vibration compensation
rq-D4.2-T.08	Robustness to dynamic variations
rq-D4.2-T.09	Set point shaping
rq-D4.2-T.10	Decoupling
rq-D4.2-T.11	Performance assessment and fault detection
rq-D4.2-T.12	Homing strategies
rq-D4.2-R.01	Model based design
rq-D4.2-R.02	Identification
rq-D4.2-R.03	Automatic tuning and adjustability
rq-D4.2-R.04	Control prototyping
rq-D4.2-R.05	Implementation

The next tables summarize the requirements coming from each pilot, demonstrator and use cases, following the defined requirement codification and providing technical details. In particular, differently from D4.1, technologies, languages and tools that will be used for the implementation of the requirements are described.

3.1 Pilot 1 (SCC) - Industrial printing - Generic substrate carrier

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-P.06	High performance amplifiers High speed communication bus	- current loop control bandwidth > 500 Hz - current loop sample rate > 32 kHz - position loop sample rate > 10 kHz - jitter < 5 µs - communication with layer 1 sample rate > 10 kHz	HW		3.5 3.6	BB1 BB5
rq-D4.2-T.02	Friction and inertia estimation and compensation	- Automatic identification of friction and inertia/mass values.	SW	Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI	5.4	BB6

rq-D4.2-T.03	Feed forward compensation	- Automatic identification of (changes in) feedforward terms: - Static friction - Viscous friction - Spring stiffness - Inertia/Mass (acceleration) - Jerk	SW	Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI	5.4	BB6
rq-D4.2-T.04	Periodic harmonic disturbance compensation	- Automatic identification of periodic harmonic disturbance: - Motor cogging - Roller eccentricity - Belt and roller manufacturing tolerances	SW	Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI	5.4	BB6
rq-D4.2-T.05	Repetitive control algorithms to provide good performance in both low and high (above feedback controller bandwidth) frequency regions.	- Algorithms shall be able to compensate for disturbances with frequencies above the bandwidth of the feedback controller.	SW	Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI	4.6	BB9
rq-D4.2-T.10	Controller shall provide robustness against the off diagonal terms (cross-couplings) for MIMO systems.	.	SW	Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI	4.5	BB8
rq-D4.2-T.11	Performance assessment and system condition monitoring	- Measurement of frequency responses (including cross-couplings for MIMO systems) - Identification of unexpected deviations (feedforward terms, friction) - Assessment of bandwidth, sensitivity, stability margins, settling time, rise time,	SW	- Monitor changes in auto-tuned parameters over time (e.g. increase in friction feedforward parameter) - Monitor deterioration of system overlay performance - Methodologies for processing and diagnostics based on measured data Programming languages: - Matlab/Simulink (control and data processing)	5.2 5.3	BB3

		overshoot, noise/disturbance levels - Assessment of performance obtained by RC algorithms		- Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI		
rq-D4.2-R.01	model-based methods for design, diagnosis and system identification		SW	Rapid (SIL, PIL, MIL) prototyping of control strategies	-	-
rq-D4.2-R.02	System identification	Automatic identification of: - System dynamics - Friction terms - Period harmonic disturbances	SW	-User-friendly system identification of FRF models SISO and MIMO Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI	5.4 4.5	BB6, BB8
rq-D4.2-R.03	Automatic tuning for all the control parameters	Automatic tuning of: -PID parameters -Bi-quadratic/Notch filters Respect position, speed, torque constraints	SW	Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI	5.4	BB6
	Online controller parameters change block	Possibility to modify online all the control parameters.	SW	Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units: - SI		
rq-D4.2-R.05	Availability of all signals	Possibility to trace and plot all signals: - Physical (measured) signals - Internal software signals	SW	Programming languages: - Matlab/Simulink (control and data processing) - Python (system automation) Test: - MIL/PIL/HIL setup Units:		BB3

				- SI		

3.2 Pilot 2 (NXP) - Semiconductor production - 12 Inch wafer stage

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-P.03	Setpoint tracking	Setpoint distance: 1mm Setpoint duration: 17ms including settling time Setpoint settling: +/-2um	SW	Achieved by automatic tuning Robust PID controller in combination with accurately tuned feedforwards	4.3, 4.5, 4.3	BB6, BB8
rq-D4.2-P.06	Control sample rate	Min: 8 kHz, preferred 20 kHz	HW/SW			BB1, BB5
rq-D4.2-T.02	models for unpredictable wafer friction and relaxation	-	SW	Not essential for performance	4.2, 4.3	
rq-D4.2-T.03	- ILC to achieve perfect setpoint tracking. - jerk/voltage feedforward in the amplifier	Implementation of ILC Feedforward control	SW	Automatic identification of models used in ILC implementation Support in tuning ILC parameters Support for mass, friction (viscous, Coulomb), stiffness, jerk, snap feedforward	4.6	BB9
rq-D4.2-T.10	decoupling considering position-dependent dynamics and thermal effects	-	SW	Not essential for performance; static decoupling may suffice	4.2, 4.3, 4.5	BB8
rq-D4.2-T.11	- system condition monitoring - drive condition monitoring	Detection of excessive friction, increase in drive current	SW	Monitoring all essential drive parameters Monitor changes in auto-tuned parameters over time (e.g. increase in friction feedforward parameter) Monitor deterioration of setpoint tracking performance Methodologies for processing and diagnostics based on measured data	5.4	BB3
rq-D4.2-R.01	model-based methods for design, diagnosis and system identification		SW	Support rapid mechanical design iterations Easy means of studying effects of non-rigid bodies and imperfect actuators Rapid (MIL) prototyping of control strategies using Sfunctions or FMI interfaces to Amesim physical models	4.2, 4.3, 4.5	-
rq-D4.2-R.02	System identification		SW	User-friendly system identification of FRF models SISO and MIMO Frequency range: selectable, usually 1-500 Hz Quantification of confidence bounds on FRF models (i.e. to design robust controllers)	4.5	BB8

rq-D4.2-R.03	- adjustable control algorithm - automatic tuning of controllers and feedforward terms - gain scheduling for position-dependent dynamics.	Automatic tuning of PID parameters and biquads for performance and robustness	SW	Automatic tuning of user-selectable feedback/feedforward structure depending on plant characteristics Support for tables and interpolation to implement gain scheduling and other compensation schemes	5.4	BB6
rq-D4.2-R.05	- support for tracing all signals, including current-loop, in real-time and for on-line changing of parameters	Online changing of PID, biquad and feedforward parameters Online changing of controller state	SW	Software scope to aid diagnosis and commissioning in conjunction with the ability to adapt parameters manually online.	4.7	BB6, BB11

3.3 Pilot 3 (IMA) - High speed packaging - In-line filling & stoppering machine, Tea bag machine

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-P.06	- cycle time: 500µs - jitter: 1% of cycle time					
rq-D4.2-R.05	- at least 100 axes - Intel x86 platform - VxWorks 6.9.x support - minimize ad-hoc source code creation in multicore platforms - efficient inter-core communication mechanisms in multicore platforms					
rq-D4.2-R.06	- specific hardware platform requirements	B&R APC910 i5 6440EQ		VT-x VT-d support, four CPU cores, at least 4GB RAM		

3.4 Pilot 4 (COR) - Big CNC - Smart machining tools and milling machines

Pilot not involved in WP4.

3.5 Pilot 5 (PHI) - Healthcare robotics - Medical manipulator

Pilot not involved in WP4. Requirements have been outlined in D4.2 because BB6, which is based on Task 5.4, is connected also to the control layer.

3.6 Demonstrator 1 (VIS) - Insulin Delivery System

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-T.10	Centralised control.	- SCADA system – Rockwell Factory Talk - Magnemotion AB PLC - Integration of AML and OPC UA – DIN SPEC 16592	HW/SW			BB11
rq-D4.2-T.11	Smart sensing and condition monitoring. Wireless sensors to monitor electro-mechanical parameters and the use of a data analysis platform. Data for predictive maintenance	- Sensor communication to Wireless AP – Ethernet IP - Wireless AP to AB PLC – IEC 61784 - Machine to machine communication – OPC UA (IEC 62541) - Industrial automation systems and integration – ISO 13584	HW/SW	- Vehicle Into Station X –Trigger RFID/IP read of UV/Temp/Energy Data - Transmit sensor ID and data to connected system (NOT PLC to start with) –Log to PC database or logfile. - Validate sensor data sent and received is correct. Include startup / shutdown / fault recovery - Validate sensor data sent and received is correct. Include startup / shutdown / fault recovery - Combine Vehicle RFID data (vehicle ID, batch ID etc transmitted to PLC) with Sensor block ID and Data - Validate: sensor ID and vehicle ID correlation is correct - Send correctly correlated Sensor/Vehicle data to SCADA database - Store contextualised data in Rockwell Factory Suite Software	5.2	BB2 BB3

3.7 Demonstrator 2 (ECS) - Injection mould tool

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-T.07	vibration data acquisition and feedback on tool during manufacturing	Sampling not yet defined, depending on the experimental results.	HW		2.3	BB2
rq-D4.2-T.11	a) Corrosion data acquisition from mould tool and predictive maintenance.	a) Sampling not yet defined for corrosion sensor, depending on the experimental results, but low	HW/SW		2.3	BB2

	b) Force sensor implementation on ejection mould side.	frequency acquisition. b) Sampling not yet defined for force sensor, depending on the experimental results.				
rq-D4.2-R0.5	Both T.07 and T.11 are still on development		HW/SW	Incorporation of both sensors of rq-D4.2-T.11 on the tool	2.3	BB2

3.8 Use case 1.1 (GEF) - Drive for industrial applications

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-T.05	implementation of repetitive control	Repetitive control block for repetitive disturbances attenuation I/O interface: - control error (I) - RC output (O) Sampling time: - 1/25 of the disturbance period Disturbance attenuation: - minimum 50% Array length for RC: - maximum 250 elements	SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Test: - Software simulation - HIL setup Units: - SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB	4.6	BB9
rq-D4.2-T.06	passive and active anti-sway strategies with or without additional sensors	Anti-sway open loop strategies (no additional sensors): - Input shaping (see rq-D4.2-T.09) - Input-output inversion (see rq-D4.2-T.09) Anti-sway closed loop strategies (additional wireless sensors): - Model predictive control Performance requested: - Residual oscillations less than 1°	SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Test: - Software simulation - HIL setup - UC1.1 Units: - SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB	4.4	BB7
rq-D4.2-T.07	iterative learning control block for vibration compensation	I/O interface: - control error (I) - ILC action (O) Vibration attenuation: - minimum 25%	SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Test: - Software simulation - HIL setup Units:	4.6	BB9

				- SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB		
rq-D4.2-T.09	input shaping	Input shaping techniques: - ZV - ZVD - ZVDD - Extra Insensitive - Extra Insensitive Two Humps - Partial Sum - Unitary Module I/O interface: - Desired cart speed set point [m/s] (I) - Shaped cart speed set point [m/s] (O) Array length for Input shaping: - Maximum 300 elements	SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Test: - Software simulation - HIL setup - UC1.1 Units: - SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB	4.6	BB7
	model inversion block	Input-output inversion techniques: - Motor Force-Payload speed (torque ff) - Cart peed-payload speed (cart speed ref) I/O interface: - Transition time [s] (I) - Steady state reference speed [m/s] (I) - Crane model parameters (I) - Cart speed set point [m/s] (O) - Cart torque ff [Nm] (O)	SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Test: - Software simulation - HIL setup - UC1.1 Units: - SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB	4.6	BB7
rq-D4.2-R.03	automatic tuning for all the control parameters	Automatic tuning of: - PID parameters - Bi-quadratic/Notch filters Respect position, speed, torque constraints	SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Test: - Software simulation - HIL setup - UC1.1 Units: - SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB		BB6

I-MECH Smart Mechatronic Solutions

	online controller parameters change block	Possibility to modify online all the control parameters: - PID parameters - Bi-quadratic/Notch filters	SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Units: - SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB		BB6
	predictive learning block		SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Units: - SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB		BB6
rq-D4.2-R.04	model-based methods for design		SW	Rapid (MIL) prototyping of control strategies integrated with Simulink physical model or using Sfunctions interfaces Amesim physical models	4.2 4.3 4.5	
rq-D4.2-R.05	availability of all signals	Possibility to trace all the signals: - Physical (measured) signals - Internal software signals	SW	Programming languages: - Matlab/Simulink - IEC 6 1131.3 Units: - SI Drives: - ADV200 series Minimum Sampling time: - 1 ms Maximum memory usage: - 60 kB		BB3

3.9 Use case 1.2 (FAG) - Compact control + HMI unit for CNC machines

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-P.06	High-performance servo amplifier	current loop bandwidth > 2kHz. current loop sample rate >= 16khz. Sample rate: 8khz (16kHz desirable) in at least one core 1kHz (4kHz desirable) in another core Full synchronization of drive and CNC loops through Ethercat	HW			BB5

rq-D4.2-T.05	repetitive control	Periodic mechanical disturbance compensation for cogging effect in electrical motors	SW	x86 platform	4.6	BB9
rq-D4.2-T.07	active vibration compensation	Load side feedback with its own kinematics Load estimation algorithms based on model Two approaches: - Active vibration based on observer - Active vibration with load acceleration feedback measured by accelerometers placed on TCP	SW	x86 platform	4.4	BB7
rq-D4.2-T.08	model-based robust control	Robust control should improve performance of a Scara Robot with: - Variable inertia dependant on TCP position - Elasticity introduced from reduction	SW	x86 platform	4.5	BB8
rq-D4.2-T.09	learning control	Using Scara Robot, a pick and place or similar application could be optimized	SW	x86 platform	4.6	BB9
rq-D4.2-T.10	multivariable control	Centralized MIMO control of at least two axes.	SW	x86 platform	4.5	BB8
rq-D4.2-R.01	Control specific multi/many core platform	Control specific multi/many core platform - 2 cores minimum, 4 cores highly desirable - x86 platform with hypervisor support	SW	Sample rate: 8Khz (16KHz desirable) in at least one core 1KHz (4KHz desirable) in another core	4.7	BB10
rq-D4.2-R.03	self-commissioning velocity and position control loops	Automatic tuning of at least standard PI controllers	SW	Other parameters related with vibration compensation or robust control could also be automatically tuned.		BB6
rq-D4.2-R.05	RTOS for multi/many core platform	Commutation method: FOC with encoder, motor position sent to CNC in ethercat. Program language: C (with intrinsic functions) General Purpose OS: Linux and Windows 7 embedded (not necessary at the same time) Hypervisor as required for "near 0" latency	SW	Full synchronization of the system loops All the parts of control loops should be accessible independently.	4.7	BB11

3.10 Use case 1.3 (TECO) - PAC based modular HW for machinery

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-P.06	Sampling rates	EtherCAT communication 1kHz, 100FPS vision system	HW SW		3.3 3.5	BB1 BB4
rq-D4.2-T.01	Signal filtering, image processing		SW		3.5	BB4
rq-D4.2-T.07	Vibration control	Compensation of oscillatory dynamics – passive and/or active stabilization	SW		4.4	BB7
rq-D4.2-R.03	Motion control loops autotuning	Automatic commissioning of velocity and position loops using experimental identification and model based controller design	SW		5.4	BB6
rq-D4.2-R.05	Implementation of control algorithms on commercial HW platform	Control algorithms will be executed on TECO PAC platform equipped with ARM CPU and RT Linux OS	SW HW		6.5	

3.11 Use case 2.1 (GMV) - Space GNC systems through the use of robotic devices

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-P.03_A	Force/impedance control set point tracking	4DOF force/impedance control. Measured magnitude: motor current sensing for force estimation. Force range < 10N Force control tolerance < 5% Impedance control: End- effector dynamics can be prescribed (compliance regarding collaborative tasks)	SW, HW	Programming languages: - C/C++ - Matlab/Simulink Test: - HIL setup Units: - SI Minimum Sampling time: - 2kHz	T3.6 T4.5	BB5 BB8
rq-D4.2-P.03_B	Visual servoing control set point tracking	Measured magnitude: XY (lateral) position of the marker, distance to marker, orientation of marker. Distance range: 0.2-1.0m Speed range: <1cm/s	SW, HW	Programming languages: - Test: - Simulation	T3.5 T4.5	BB4 BB8

		Vision control tolerance: 2mm (at 0.2m)		- HIL setup Units: - SI Minimum Sampling time: - 5Hz		
rq-D4.2-P.06_A	Force/impedance control Bandwidth and sampling frequency	Force control loop, 4axis: 500Hz (2ms) Current loop bandwidth > 2kHz. Current loop sample rate > 16khz. EtherCAT communication period 16 kHz	SW, HW	Programming languages: - C/C++ - Matlab/Simulink Test: - HIL setup Units: - SI Minimum Sampling time: - 2kHz	T3.6 T4.5	BB5 BB8
rq-D4.2-P.06_B	Visual servoing control set point tracking	Bandwidth per camera: - actual: nominal 150-200 Mbps - desirable up to 400 Mbps - 800 Mbps Vision control loop > 5Hz (200ms), Desirable 50Hz (20ms)	SW, HW	Programming languages: - Test: - Simulation - HIL setup Units: - SI Minimum Sampling time: - 5 Hz	T3.5 T4.5	BB4 BB8
rq-D4.2-T.09	Set point shaping	- trajectory generation, inverse kinematics - learning control.	SW	Programming languages: - C/C++ - Matlab/Simulink Test: - HIL setup Units: - SI Minimum Sampling time: - 2kHz	T4.5	BB8
rq-D4.2-T.10	Force/impedance control coupling	4 MIMO Axes force/impedance. (Desirable 6 axes).	SW	Programming languages: - C/C++ - Matlab/Simulink Test: - HIL setup Units: - SI Minimum Sampling time:	T4.5	BB8

				- 2kHz		
Rq-D4.2-R.04	Deployment	Implementation of the motion control for the force/vision-controlled device in a separate hardware allowing an easy and fast deployment and testing on the platform-art©	HW		T3.5	BB4
Rq-D4.2-R.05		Interface (e.g., Cameralikn, USB3, GiGE, Firewire 800..): - Actual: SpaceWire, GiGE, USB3.0, Camera-Link - Desired: + SpaceFibre and TSN/TTEthernet Number of realtime streaming cameras: - Actual: 1 - Desired: 2 Resolution: - Actual: 2048x2048 10-12bits and 1024x1024 10-12bits - Desired: 2048x2048 10-12bits and 1024x1024 10-12bits	HW		T3.5	BB4

3.12 Use case 2.2 (ZAPUNI) - Open modular robotic arm

Code	Description	Technical details	Type	Technologies, languages and tools	Task	BB
rq-D4.2-P.03	Force control algorithms	Force control: 6 DoF (end effector F/T sensor), sensitivity < 1 N Impedance control: End-effector dynamics can be prescribed (compliance regarding collaborative tasks)	SW		T4.5	BB8
rq-D4.2-P.06	Sampling rates	EtherCAT 2kHz, 7 axes	SW		T3.3	BB1
rq-D4.2-T.10	Multivariable control	MIMO 7 axes Inverse kinematics algorithms: - Default: direct control of redundant axis:	SW		T4.5	BB8

		- advanced: joint limits/velocities limitation, obstacle overcoming, singularity overcoming				
rq-D4.2-T.12	Homing procedure	robot homing via machine vision system	SW		T4.5	BB8
rq-D4.2-R.03	Autotuning	Tuning of individual motion control loops	SW		T5.4	BB6
rq-D4.2-R.05	Implementation	Payload 1 kg Reach < 1 m Weight < 20kg Applications: Pick & Place, Gluing, Small part assembly, screwing, etc.	HW SW			

4 Common protocols, interfaces, languages and tools

The following table summarizes the specific requirements of Pilot, Use Cases and Demonstrators regarding the use of the protocols, interfaces, languages and tools outlined in D2.4 for the I-MECH reference platform.

	Pilot 1	Pilot 2	Pilot 3	Pilot 4	Pilot 5	Demo 1	Demo 2	Use Case 1.1	Use Case 1.2	Use Case 1.3	Use Case 2.1	Use Case 2.2
Protocols												
EtherCAT	X	X						X	X	X	X	X
Wireless						X		X				
Ethernet						X					X	
Interfaces												
OPC UA						X						
FMI		X										
Languages												
Matlab/Simulink	X	X						X			X	
C/C++									X		X	
Tools												
Matlab/Simulink	X	X						X			X	
Amesim		X						X				

5 Building Blocks developments

The BBs (in particular, here we are referring to those related to the control methodologies, namely BB6-BB9) should contain the methodologies that will meet the least common denominator in terms of pilots, demonstrators, and use cases requirements, so that they will form (together with the other BBs) the basis of the I-MECH platform. In this context, the BBs owners (in a bottom up approach) are requested to conceive;
- how the creation process of the BBs will satisfy this general requirement;
- how the final BBs will fit into the I-MECH platform (that is, how the BBs will satisfy the requirements of D2.4)
- how their validation and testing will be performed (see D6.2)
- how they can finally be customized by the pilot owners in order to satisfy their specific requirements.

It is worth stressing that fundamental tool for the development of the Building Blocks and for their validation is Matlab/Simulink, which is the de facto standard environment for control systems design and development and it is employed by the great majority of I-MECH partners. It can be also used together with other modelling tools like Amesim. Once the building blocks have been created in this framework, the pilots, demonstrators and use cases owners will be able to use them by exploiting the code generation function of Simulink. In the future, this should also allow other users outside the I-MECH consortium to exploit the I-MECH platform to implement advanced control methodologies in an easier way.

6 Conclusions

In this deliverable, final requirements and specifications for I-MECH motion control applications are stated. Each specification has been detailed and connected to the corresponding task(s) and building block(s). This deliverable will be exploited in WP6 for the verification and validation phase of the building blocks and in WP7 in order to verify the effectiveness of the I-MECH platform in the pilots.

Acknowledgement
This project has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737453. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation program and Netherlands, Czech Republic, Latvia, Spain, Greece, Portugal, Belgium, Italy, France, Ireland.