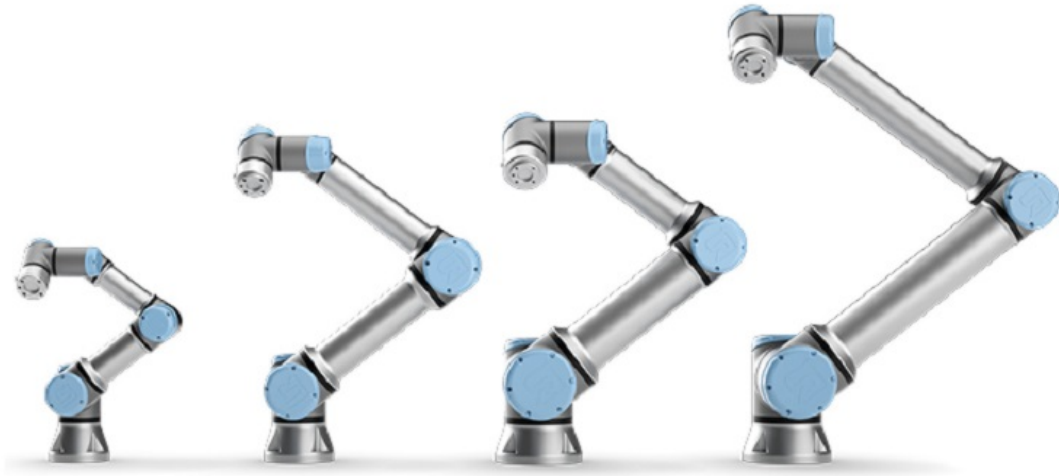




UNIVERSAL ROBOTS



Primary/Secondary Client Interface

The information contained herein is the property of Universal Robots A/S and shall not be reproduced in whole or in part without prior written approval of Universal Robots A/S. The information herein is subject to change without notice and should not be construed as a commitment by Universal Robots A/S. This document is periodically reviewed and revised.

Universal Robots A/S assumes no responsibility for any errors or omissions in this document.

Copyright © 2009–2021 by Universal Robots A/S.

The Universal Robots logo is a registered trademark of Universal Robots A/S.

Contents

1. Introduction	1
2. Robot Software 5.9	2
2.1. Messages sent to primary and secondary clients	2
2.1.1. Robot State Message	2
2.1.2. Robot mode data (Sub Package of Robot State Message)	2
2.1.3. Joint data (Sub Package of Robot State Message)	3
2.1.4. Tool data (Sub Package of Robot State Message)	3
2.1.5. Masterboard data (Sub Package of Robot State Message)	4
2.1.6. Cartesian info (Sub Package of Robot State Message)	5
2.1.7. Kinematics info (Sub Package of Robot State Message)	5
2.1.8. Configuration data (Sub Package of Robot State Message)	6
2.1.9. Force mode data (Sub Package of Robot State Message)	7
2.1.10. Additional info (Sub Package of Robot State Message)	7
2.1.11. Calibration data (Sub Package of Robot State Message, Internally used only)	8
2.1.12. Safety Data (Sub Package of Robot State Message)	8
2.1.13. Tool Communication Info (Sub Package of Robot State Message)	8
2.1.14. Tool Mode Info (Sub Package of Robot State Message)	9
2.1.15. Singularity Info (Sub Package of Robot State Message)	9
2.1.16. Version Message (sent only once)	9
2.2. Messages sent to primary clients ONLY	10
2.2.1. Robot Message - Safety Mode Message	10
2.2.2. Robot Message - Robot Comm Message	10
2.2.3. Robot Message - Key Message	10
2.2.4. Robot Message - Program Threads Message	11
2.2.5. Robot Message - Popup Message	11
2.2.6. Robot Message - Request Value Message	12
2.2.7. Robot Message - Text Message	12
2.2.8. Robot Message - Runtime Exception Message	12
2.2.9. Global Variables Setup Message	13
2.2.10. Global Variables Update Message	13
3. Robot Software 5.10	14
3.1. Messages sent to primary and secondary clients	14
3.1.1. Robot State Message	14
3.1.2. Robot mode data (Sub Package of Robot State Message)	14
3.1.3. Joint data (Sub Package of Robot State Message)	15



3.1.4. Tool data (Sub Package of Robot State Message)	15
3.1.5. Masterboard data (Sub Package of Robot State Message)	16
3.1.6. Cartesian info (Sub Package of Robot State Message)	17
3.1.7. Kinematics info (Sub Package of Robot State Message)	17
3.1.8. Configuration data (Sub Package of Robot State Message)	18
3.1.9. Force mode data (Sub Package of Robot State Message)	19
3.1.10. Additional info (Sub Package of Robot State Message)	19
3.1.11. Calibration data (Sub Package of Robot State Message, Internally used only)	20
3.1.12. Safety Data (Sub Package of Robot State Message)	20
3.1.13. Tool Communication Info (Sub Package of Robot State Message)	20
3.1.14. Tool Mode Info (Sub Package of Robot State Message)	21
3.1.15. Singularity Info (Sub Package of Robot State Message)	21
3.1.16. Version Message (sent only once)	21
3.2. Messages sent to primary clients ONLY	22
3.2.1. Robot Message - Safety Mode Message	22
3.2.2. Robot Message - Robot Comm Message	22
3.2.3. Robot Message - Key Message	22
3.2.4. Robot Message - Program Threads Message	23
3.2.5. Robot Message - Popup Message	23
3.2.6. Robot Message - Request Value Message	24
3.2.7. Robot Message - Text Message	24
3.2.8. Robot Message - Runtime Exception Message	24
3.2.9. Global Variables Setup Message	25
3.2.10. Global Variables Update Message	25
4. Control Modes	26
5. Robot Modes	27
6. Joint Modes	28
7. Tool Modes	29
8. Message Sources	30
9. Safety Mode Types	32
10. Safety Status Types	33
11. Report Levels	34
12. Requested Types	35
13. Value Types	36

1. Introduction

The Primary Client Interface (10 Hz)

The information on this section applies to UR software versions shown, below:

Primary client=port 30001 (Robot state and messages)

Secondary client=port 30002 (Only robot state and the Version messages)

Primary client **read only**=port 30011 (Robot state and messages)

Secondary client **read only**=port 30012 (Only robot state and the Version messages)

2. Robot Software 5.9

Message formatting

4 bytes (int)	Length of overall package
1 byte (uchar)	Robot MessageType
4 bytes (int)	Length of Sub-Package
1 byte (uchar)	Package-Type
n bytes	Content...
4 bytes	Length of Sub-Package
1 byte	Package-Type
n bytes	Content...
	...

2.1. Messages sent to primary and secondary clients

Messages not documented in this manual are only used internally by UR software and are not backwards compatible

2.1.1. Robot State Message

int messageSize	Length of overall package: total length of package including this field
unsigned char messageType = MESSAGE_TYPE_ROBOT_STATE = 16	
Sub packages	Each sub package that follows contains size, type, and package specific data

2.1.2. Robot mode data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_ROBOT_MODE_DATA = 0	

uint64_t timestamp	
bool isRealRobotConnected	
bool isRealRobotEnabled	
bool isRobotPowerOn	
bool isEmergencyStopped	
bool isProtectiveStopped	
bool isProgramRunning	
bool isProgramPaused	
unsigned char robotMode	Robot Modes
unsigned char controlMode	Control Modes
double targetSpeedFraction	
double speedScaling	
double targetSpeedFractionLimit	
unsigned char reserved	

2.1.3. Joint data (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_JOINT_DATA = 1	
for each joint:	
double q_actual	
double q_target	
double qd_actual	
float l_actual	
float V_actual	
float T_motor	
float T_micro	Deprecated - ignore
uint8_t jointMode	Joint Modes
end	

The "jointMode" field is a code for the joint status (shown on the initialisation screen):

2.1.4. Tool data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
-----------------	--

unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_TOOL_DATA = 2	
unsigned char analogInputRange0	
unsigned char analogInputRange1	
double analogInput0	
double analogInput1	
float toolVoltage48V	
unsigned char toolOutputVoltage	
float toolCurrent	
float toolTemperature	
uint8_t toolMode	Tool Modes

The "toolMode" field is a code for the joint status (shown on the initialisation screen):

2.1.5. Masterboard data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_MASTERBOARD_DATA = 3	
int digitalInputBits	
int digitalOutputBits	
unsigned char analogInputRange0	
unsigned char analogInputRange1	
double analogInput0	
double analogInput1	
char analogOutputDomain0	
char analogOutputDomain1	
double analogOutput0	
double analogOutput1	
float masterBoardTemperature	
float robotVoltage48V	
float robotCurrent	
float masterIOCurrent	
unsigned char safetyMode	Safety Modes
uint8_t InReducedMode	
char euromap67InterfaceInstalled	

if euromap67 interface is installed, also the following:	
uint32_t euromapInputBits	
uint32_t euromapOutputBits	
float euromapVoltage24V	
float euromapCurrent	
end	
uint32_t (Used by Universal Robots software only)	
uint8_t operationalModeSelectorInput	
uint8_t threePositionEnablingDeviceInput	
unsigned char (Used by Universal Robots software only)	

2.1.6. Cartesian info (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_CARTESIAN_INFO = 4	
double X	
double Y	
double Z	
double Rx	
double Ry	
double Rz	
double TCPOffsetX	
double TCPOffsetY	
double TCPOffsetZ	
double TCPOffsetRx	
double TCPOffsetRy	
double TCPOffsetRz	

2.1.7. Kinematics info (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_KINEMATICS_INFO = 5	

for each joint:
uint32_t cheksum
for each joint:
double DHtheta
for each joint:
double DHa
for each joint:
double Dh
for each joint:
double Dalpha
uint32_t calibration_status

This information is sent when leaving initializing mode and/or if the kinematics configuration is changed.

2.1.8. Configuration data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_CONFIGURATION_DATA = 6	
for each joint:	
double jointMinLimit	
double jointMaxLimitt	
end	
for each joint:	
double jointMaxSpeed	
double jointMaxAcceleration	
end	
double vJointDefault	
double aJointDefault	
double vToolDefault	
double aToolDefault	
double eqRadius	
for each joint:	
double DHa	
for each joint:	
double Dh _d	

for each joint:	
double DHalpha	
for each joint:	
double DHtheta	
int32_t motherboardVersion	
int32_t controllerBoxType	
int32_t robotType	
int32_t robotSubType	

This information is sent when leaving initializing mode and/or if the kinematics configuration is changed.

2.1.9. Force mode data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_FORCE_MODE_DATA = 7	
double Fx	
double Fy	
double Fz	
double Frx	
double Fry	
double Frz	
double robotDexterity	

2.1.10. Additional info (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_ADDITIONAL_INFO = 8	
unsigned char tpButtonState	
bool freedriveButtonEnabled	
bool IOEnabledFreedrive	
unsigned char reserved	

2.1.11. Calibration data (Sub Package of Robot State Message, Internally used only)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_NEEDED_FOR_CALIB_DATA = 9	
double Fx	
double Fy	
double Fz	
double Frx	
double Fry	
double Frz	

This package is used internally by Universal Robots software only and should be skipped.

2.1.12. Safety Data (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = 10	
safety data	

It is used internally by Universal Robots software only and should be skipped.

2.1.13. Tool Communication Info (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_TOOL_COMM_INFO = 11	
bool toolCommunicationIsEnabled	
int32_t baudRate	
int32_t parity	
int32_t stopBits	
float RxIdleChars	
float TxIdleChars	

2.1.14. Tool Mode Info (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_TOOL_MODE_INFO = 12	
uint8_t output mode	
uint8_t digital output mode output 0	
uint8_t digital output mode output 1	

2.1.15. Singularity Info (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_SINGULARITY_INFO = 13	
uint8_t singularitySeverity	
uint8_t singularityType	
It is used internally by Universal Robots software only and should be skipped.	

2.1.16. Version Message (sent only once)

This is the first package sent on both the primary and secondary client interfaces. This package it is not part of the robot state message.

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_VERSION = 3	
char projectNameSize	
charArray projectName	
unsigned char majorVersion	
unsigned char minorVersion	
int bugfixVersion	
int buildNumber	
charArray buildDate	

2.2. Messages sent to primary clients ONLY

2.2.1. Robot Message - Safety Mode Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_SAFETY_MODE = 5	
int robotMessageCode	
int robotMessageArgument	
unsigned char safetyModeType	Safety Mode Types
uint32_t reportDataType	
uint32_t reportData	

2.2.2. Robot Message - Robot Comm Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_ERROR_CODE = 6	
int robotMessageCode	
int robotMessageArgument	
int robotMessageReportLevel	Report Levels
uint32_t robotMessageDataType	
uint32_t robotMessageData	
charArray robotCommTextMessage	

2.2.3. Robot Message - Key Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources

char robotMessageType = ROBOT_MESSAGE_TYPE_KEY = 7	
int robotMessageCode	
int robotMessageArgument	
uint8_t robotMessageTitleSize	
charArray robotMessageTitle	
charArray keyTextMessage	

2.2.4. Robot Message - Program Threads Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_PROGRAM_LABEL_THREADS = 14	
following structure repeats for each running program thread	
int labelId (line number)	
int labelNameLength	
charArray labelName	
int threadNameLength	
charArray threadName	

2.2.5. Robot Message - Popup Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_POPUP = 2	
unsigned int requestId	
unsigned int requestedType	Requested Types
bool warning	
bool error	
bool blocking	
uint8_t popupMessageTitleSize	
charArray popupMessageTitle	



charArray popupTextMessage	
----------------------------	--

2.2.6. Robot Message - Request Value Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_REQUEST_VALUE = 9	
unsigned int requestId	
unsigned int requestedType	Requested Types
char array requestTextMessage	

2.2.7. Robot Message - Text Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_TEXT = 0	
charArray textTextMessage	

2.2.8. Robot Message - Runtime Exception Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_RUNTIME_EXCEPTION = 10	
int scriptLineNumber	
int scriptColumnNumber	

charArray runtimeExceptionTextMessage

2.2.9. Global Variables Setup Message

int messageSize	
unsigned char messageType = MESSAGE_	
TYPE_PROGRAM_STATE_MESSAGE = 25	
uint64_t timestamp	
char robotMessageType = PROGRAM_STATE_	
MESSAGE_TYPE_GLOBAL_VARIABLES_SETUP	
= 0	
uint16_t startIndex	
charArray variableNames	List of names separated by new line
	character ('\n' character). Example: var_
	1\nvar_2\n

2.2.10. Global Variables Update Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_PROGRAM_STATE_MESSAGE =	
25	
uint64_t timestamp	
char robotMessageType = PROGRAM_STATE_MESSAGE_TYPE_GLOBAL_	
VARIABLES_UPDATE = 1	
uint16_t startIndex	
unsigned char variableValues	Value
	Types

Each variable value contains type byte, data, and terminating new line character (\n character).

Example - two variables of type **BOOL(True), and **STRING("aaaa")**:** 0c 01 0a03 00 04 61 61 61 61 0a

3. Robot Software 5.10

Message formatting

4 bytes (int)	Length of overall package
1 byte (uchar)	Robot MessageType
4 bytes (int)	Length of Sub-Package
1 byte (uchar)	Package-Type
n bytes	Content...
4 bytes	Length of Sub-Package
1 byte	Package-Type
n bytes	Content...
	...

3.1. Messages sent to primary and secondary clients

Messages not documented in this manual are only used internally by UR software and are not backwards compatible

3.1.1. Robot State Message

int messageSize	Length of overall package: total length of package including this field
unsigned char messageType = MESSAGE_TYPE_ROBOT_STATE = 16	
Sub packages	Each sub package that follows contains size, type, and package specific data

3.1.2. Robot mode data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_ROBOT_MODE_DATA = 0	

uint64_t timestamp	
bool isRealRobotConnected	
bool isRealRobotEnabled	
bool isRobotPowerOn	
bool isEmergencyStopped	
bool isProtectiveStopped	
bool isProgramRunning	
bool isProgramPaused	
unsigned char robotMode	Robot Modes
unsigned char controlMode	Control Modes
double targetSpeedFraction	
double speedScaling	
double targetSpeedFractionLimit	
unsigned char reserved	

3.1.3. Joint data (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_JOINT_DATA = 1	
for each joint:	
double q_actual	
double q_target	
double qd_actual	
float l_actual	
float V_actual	
float T_motor	
float T_micro	Deprecated - ignore
uint8_t jointMode	Joint Modes
end	

The "jointMode" field is a code for the joint status (shown on the initialisation screen):

3.1.4. Tool data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
-----------------	--

unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_TOOL_DATA = 2	
unsigned char analogInputRange0	
unsigned char analogInputRange1	
double analogInput0	
double analogInput1	
float toolVoltage48V	
unsigned char toolOutputVoltage	
float toolCurrent	
float toolTemperature	
uint8_t toolMode	Tool Modes

The "toolMode" field is a code for the joint status (shown on the initialisation screen):

3.1.5. Masterboard data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_MASTERBOARD_DATA = 3	
int digitalInputBits	
int digitalOutputBits	
unsigned char analogInputRange0	
unsigned char analogInputRange1	
double analogInput0	
double analogInput1	
char analogOutputDomain0	
char analogOutputDomain1	
double analogOutput0	
double analogOutput1	
float masterBoardTemperature	
float robotVoltage48V	
float robotCurrent	
float masterIOCurrent	
unsigned char safetyMode	Safety Modes
uint8_t InReducedMode	
char euromap67InterfaceInstalled	

if euromap67 interface is installed, also the following:	
uint32_t euromapInputBits	
uint32_t euromapOutputBits	
float euromapVoltage24V	
float euromapCurrent	
end	
uint32_t (Used by Universal Robots software only)	
uint8_t operationalModeSelectorInput	
uint8_t threePositionEnablingDeviceInput	
unsigned char (Used by Universal Robots software only)	

3.1.6. Cartesian info (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_CARTESIAN_INFO = 4	
double X	
double Y	
double Z	
double Rx	
double Ry	
double Rz	
double TCPOffsetX	
double TCPOffsetY	
double TCPOffsetZ	
double TCPOffsetRx	
double TCPOffsetRy	
double TCPOffsetRz	

3.1.7. Kinematics info (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_KINEMATICS_INFO = 5	



for each joint:
uint32_t cheksum
for each joint:
double DHtheta
for each joint:
double DHa
for each joint:
double Dh
for each joint:
double Dalpha
uint32_t calibration_status

This information is sent when leaving initializing mode and/or if the kinematics configuration is changed.

3.1.8. Configuration data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_CONFIGURATION_DATA = 6	
for each joint:	
double jointMinLimit	
double jointMaxLimitt	
end	
for each joint:	
double jointMaxSpeed	
double jointMaxAcceleration	
end	
double vJointDefault	
double aJointDefault	
double vToolDefault	
double aToolDefault	
double eqRadius	
for each joint:	
double DHa	
for each joint:	
double Dh	

for each joint:	
double DHalpha	
for each joint:	
double DHtheta	
int32_t motherboardVersion	
int32_t controllerBoxType	
int32_t robotType	
int32_t robotSubType	

This information is sent when leaving initializing mode and/or if the kinematics configuration is changed.

3.1.9. Force mode data (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_FORCE_MODE_DATA = 7	
double Fx	
double Fy	
double Fz	
double Frx	
double Fry	
double Frz	
double robotDexterity	

3.1.10. Additional info (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_ADDITIONAL_INFO = 8	
unsigned char tpButtonState	
bool freedriveButtonEnabled	
bool IOEnabledFreedrive	
unsigned char reserved	

3.1.11. Calibration data (Sub Package of Robot State Message, Internally used only)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_NEEDED_FOR_CALIB_DATA = 9	
double Fx	
double Fy	
double Fz	
double Frx	
double Fry	
double Frz	

This package is used internally by Universal Robots software only and should be skipped.

3.1.12. Safety Data (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = 10	
safety data	

It is used internally by Universal Robots software only and should be skipped.

3.1.13. Tool Communication Info (Sub Package of Robot State Message)

int packageSize	Length of Sub Package: total length of subpackage including this field
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_TOOL_COMM_INFO = 11	
bool toolCommunicationIsEnabled	
int32_t baudRate	
int32_t parity	
int32_t stopBits	
float RxIdleChars	
float TxIdleChars	

3.1.14. Tool Mode Info (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_TOOL_MODE_INFO = 12	
uint8_t output mode	
uint8_t digital output mode output 0	
uint8_t digital output mode output 1	

3.1.15. Singularity Info (Sub Package of Robot State Message)

int packageSize	
unsigned char packageType = ROBOT_STATE_PACKAGE_TYPE_SINGULARITY_INFO = 13	
uint8_t singularitySeverity	
uint8_t singularityType	
It is used internally by Universal Robots software only and should be skipped.	

3.1.16. Version Message (sent only once)

This is the first package sent on both the primary and secondary client interfaces. This package it is not part of the robot state message.

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_VERSION = 3	
char projectNameSize	
charArray projectName	
unsigned char majorVersion	
unsigned char minorVersion	
int bugfixVersion	
int buildNumber	
charArray buildDate	

3.2. Messages sent to primary clients ONLY

3.2.1. Robot Message - Safety Mode Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_SAFETY_MODE = 5	
int robotMessageCode	
int robotMessageArgument	
unsigned char safetyModeType	Safety Mode Types
uint32_t reportDataType	
uint32_t reportData	

3.2.2. Robot Message - Robot Comm Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_ERROR_CODE = 6	
int robotMessageCode	
int robotMessageArgument	
int robotMessageReportLevel	Report Levels
uint32_t robotMessageDataType	
uint32_t robotMessageData	
charArray robotCommTextMessage	

3.2.3. Robot Message - Key Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources

char robotMessageType = ROBOT_MESSAGE_TYPE_KEY = 7	
int robotMessageCode	
int robotMessageArgument	
uint8_t robotMessageTitleSize	
charArray robotMessageTitle	
charArray keyTextMessage	

3.2.4. Robot Message - Program Threads Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_PROGRAM_LABEL_THREADS = 14	
following structure repeats for each running program thread	
int labelId (line number)	
int labelNameLength	
charArray labelName	
int threadNameLength	
charArray threadName	

3.2.5. Robot Message - Popup Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_POPUP = 2	
unsigned int requestId	
unsigned int requestedType	Requested Types
bool warning	
bool error	
bool blocking	
uint8_t popupMessageTitleSize	
charArray popupMessageTitle	



charArray popupTextMessage	
----------------------------	--

3.2.6. Robot Message - Request Value Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_REQUEST_VALUE = 9	
unsigned int requestId	
unsigned int requestedType	Requested Types
char array requestTextMessage	

3.2.7. Robot Message - Text Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_TEXT = 0	
charArray textTextMessage	

3.2.8. Robot Message - Runtime Exception Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_ROBOT_MESSAGE = 20	
uint64_t timestamp	
char source	Message Sources
char robotMessageType = ROBOT_MESSAGE_TYPE_RUNTIME_EXCEPTION = 10	
int scriptLineNumber	
int scriptColumnNumber	

charArray runtimeExceptionTextMessage

3.2.9. Global Variables Setup Message

int messageSize	
unsigned char messageType = MESSAGE_	
TYPE_PROGRAM_STATE_MESSAGE = 25	
uint64_t timestamp	
char robotMessageType = PROGRAM_STATE_	
MESSAGE_TYPE_GLOBAL_VARIABLES_SETUP	
= 0	
uint16_t startIndex	
charArray variableNames	List of names separated by new line
	character ('\n' character). Example: var_
	1\nvar_2\n

3.2.10. Global Variables Update Message

int messageSize	
unsigned char messageType = MESSAGE_TYPE_PROGRAM_STATE_MESSAGE =	
25	
uint64_t timestamp	
char robotMessageType = PROGRAM_STATE_MESSAGE_TYPE_GLOBAL_	
VARIABLES_UPDATE = 1	
uint16_t startIndex	
unsigned char variableValues	Value
	Types

Each variable value contains type byte, data, and terminating new line character (\n character).

Example - two variables of type **BOOL(True)**, and **STRING("aaaa")**: 0c 01 0a03 00 04 61 61 61 61 0a



4. Control Modes

Control Modes	
Mode	Description
0	CONTROL_MODE_POSITION
1	CONTROL_MODE_TEACH
2	CONTROL_MODE_FORCE
3	CONTROL_MODE_TORQUE

5. Robot Modes

Robot Modes	
Mode	Description
-1	ROBOT_MODE_NO_CONTROLLER
0	ROBOT_MODE_DISCONNECTED
1	ROBOT_MODE_CONFIRM_SAFETY
2	ROBOT_MODE_BOOTING
3	ROBOT_MODE_POWER_OFF
4	ROBOT_MODE_POWER_ON
5	ROBOT_MODE_IDLE
6	ROBOT_MODE_BACKDRIVE
7	ROBOT_MODE_RUNNING
8	ROBOT_MODE_UPDATING_FIRMWARE



6. Joint Modes

	Value
JOINT_MODE_RESET	235
JOINT_MODE_SHUTTING_DOWN	236
JOINT_MODE_BACKDRIVE	238
JOINT_MODE_POWER_OFF	239
JOINT_MODE_READY_FOR_POWER_OFF (FROM VERSION 5.1)	240
JOINT_MODE_NOT_RESPONDING	245
JOINT_MODE_MOTOR_INITIALISATION	246
JOINT_MODE_BOOTING	247
JOINT_MODE_VIOLATION	251
JOINT_MODE_FAULT	252
JOINT_MODE_RUNNING	253
JOINT_MODE_IDLE	255

7. Tool Modes

	Value
JOINT_MODE_RESET	235
JOINT_MODE_SHUTTING_DOWN	236
JOINT_MODE_POWER_OFF	239
JOINT_MODE_NOT_RESPONDING	245
JOINT_MODE_BOOTING	247
JOINT_MODE_BOOTLOADER	249
JOINT_MODE_FAULT	252
JOINT_MODE_RUNNING	253
JOINT_MODE_IDLE	255

8. Message Sources

Each message sent has a "source" code for the sender of the message.

	Value
MESSAGE_SOURCE_JOINT_0_FPGA	100
MESSAGE_SOURCE_JOINT_0_A	110
MESSAGE_SOURCE_JOINT_0_B	120
MESSAGE_SOURCE_JOINT_1_FPGA	101
MESSAGE_SOURCE_JOINT_1_A	111
MESSAGE_SOURCE_JOINT_1_B	121
MESSAGE_SOURCE_JOINT_2_FPGA	102
MESSAGE_SOURCE_JOINT_2_A	112
MESSAGE_SOURCE_JOINT_2_B	122
MESSAGE_SOURCE_JOINT_3_FPGA	103
MESSAGE_SOURCE_JOINT_3_A	113
MESSAGE_SOURCE_JOINT_3_B	123
MESSAGE_SOURCE_JOINT_4_FPGA	104
MESSAGE_SOURCE_JOINT_4_A	114
MESSAGE_SOURCE_JOINT_4_B	124
MESSAGE_SOURCE_JOINT_5_FPGA	105
MESSAGE_SOURCE_JOINT_5_A	115
MESSAGE_SOURCE_JOINT_5_B	125
MESSAGE_SOURCE_TOOL_FPGA	106
MESSAGE_SOURCE_TOOL_A	116
MESSAGE_SOURCE_TOOL_B	126
MESSAGE_SOURCE_EUROMAP_FPGA	107
MESSAGE_SOURCE_EUROMAP_A	117
MESSAGE_SOURCE_EUROMAP_B	127
MESSAGE_SOURCE_TEACH_PENDANT_A	108
MESSAGE_SOURCE_TEACH_PENDANT_B	118
MESSAGE_SOURCE_SCB_FPGA	40
MESSAGE_SAFETY_PROCESSOR_UA	20
MESSAGE_SAFETY_PROCESSOR_UB	30
MESSAGE_SOURCE_ROBOTINTERFACE	-2
MESSAGE_SOURCE_RTMACHINE	-3

MESSAGE_SOURCE_SIMULATED_ROBOT	-4
MESSAGE_SOURCE_GUI	-5
MESSAGE_SOURCE_CONTROLLER	7
MESSAGE_SOURCE_RTDE	8
The message types are:	
MESSAGE_TYPE_DISCONNECT	-1
MESSAGE_TYPE_ROBOT_STATE	16
MESSAGE_TYPE_ROBOT_MESSAGE	20
MESSAGE_TYPE_HMC_MESSAGE	22
MESSAGE_TYPE_MODBUS_INFO_MESSAGE	5
MESSAGE_TYPE_SAFETY_SETUP_BROADCAST_MESSAGE	23
MESSAGE_TYPE_SAFETY_COMPLIANCE_TOLERANCES_MESSAGE	24
MESSAGE_TYPE_PROGRAM_STATE_MESSAGE	25

9. Safety Mode Types

	Value	Comment
SAFETY_MODE_UNDEFINED_ SAFETY_MODE	11	
SAFETY_MODE_VALIDATE_ JOINT_ID	10	
SAFETY_MODE_FAULT	9	
SAFETY_MODE_VIOLATION	8	
SAFETY_MODE_ROBOT_ EMERGENCY_STOP	7	(EA + EB + SBUS->Euromap67) Physical e-stop interface input activated
SAFETY_MODE_SYSTEM_ EMERGENCY_STOP	6	(EA + EB + SBUS->Screen) Physical e-stop interface input activated
SAFETY_MODE_SAFEGUARD_ STOP	5	(SI0 + SI1 + SBUS) Physical s-stop interface input
SAFETY_MODE_RECOVERY	4	
SAFETY_MODE_PROTECTIVE_ STOP	3	
SAFETY_MODE_REDUCED	2	
SAFETY_MODE_NORMAL	1	

10. Safety Status Types

	Value	Comment
SAFETY_STATUS_SYSTEM_THREE_POSITION_ENABLING_STOP	13	
SAFETY_STATUS_AUTOMATIC_MODE_SAFEGUARD_STOP	12	
SAFETY_STATUS_UNDEFINED_SAFETY_MODE	11	
SAFETY_STATUS_VALIDATE_JOINT_ID	10	
SAFETY_STATUS_FAULT	9	
SAFETY_STATUS_VIOLATION	8	
SAFETY_STATUS_ROBOT_EMERGENCY_STOP	7	(EA + EB + SBUS->Euromap67) Physical e-stop interface input activated
SAFETY_STATUS_SYSTEM_EMERGENCY_STOP	6	(EA + EB + SBUS->Screen) Physical e-stop interface input activated
SAFETY_STATUS_SAFEGUARD_STOP	5	(SI0 + SI1 + SBUS) Physical s-stop interface input
SAFETY_STATUS_RECOVERY	4	
SAFETY_STATUS_PROTECTIVE_STOP	3	
SAFETY_STATUS_REDUCED	2	
SAFETY_STATUS_NORMAL	1	



11. Report Levels

	Value
REPORT_LEVEL_DEBUG (INTERNAL USE ONLY)	0
REPORT_LEVEL_INFO	1
REPORT_LEVEL_WARNING	2
REPORT_LEVEL_VIOLATION	3
REPORT_LEVEL_FAULT	4
REPORT_LEVEL_DEVL_DEBUG (INTERNAL USE ONLY)	128
REPORT_LEVEL_DEVL_INFO	129
REPORT_LEVEL_DEVL_WARNING	130
REPORT_LEVEL_DEVL_VIOLATION	131
REPORT_LEVEL_DEVL_FAULT	132

12. Requested Types

	Value
REQUEST_VALUE_TYPE_BOOLEAN	0
REQUEST_VALUE_TYPE_INTEGER	1
REQUEST_VALUE_TYPE_FLOAT	2
REQUEST_VALUE_TYPE_STRING	3
REQUEST_VALUE_TYPE_POSE	4
REQUEST_VALUE_TYPE_JOINTVECTOR	5
REQUEST_VALUE_TYPE_WAYPOINT (UNUSED)	6
REQUEST_VALUE_TYPE_EXPRESSION (UNUSED)	7
REQUEST_VALUE_TYPE_NONE (*)	8

(*) If the 'requestedType' is set to the value 8, then it is a 'PopupMessage' type.



13. Value Types

NONE

// No content -> has not been initialized yet

CONST_VAR_STRING

int16_t valueLength

charArray value

VAR_STRING

int16_t valueLength

charArray value

LIST

int16_ listLength

For each item:

uint8_t valueType

DataValueType

end

POSE

float X

float Y

float Z

float Rx

float Ry

float Rz

BOOL

bool value

NUM

float value

INT

int32_t value
FLOAT
float value
MATRIX
int16_t nRows
int16_t nColumns
For each value in each row:
uint8_t valueType
DataValueType
end

Data sizes	
int16_t	2 bytes
int32_t	4 bytes
float	4 bytes
double	8 bytes
bool	1 byte
uint8_t	1 byte

Up to software 3.2
NONE = 0
STRING = 3
LIST = 4
POSE = 9
BOOL = 11
NUM = 12
INT = 13
FLOAT = 14

From software 3.3/5.0
NONE = 0
CONST_STRING = 3



VAR_STRING = 4

LIST = 5

POSE = 10

BOOL = 12

NUM = 13

INT = 14

FLOAT = 15

Starting with software 5.9

NONE = 0

CONST_STRING = 3

VAR_STRING = 4

POSE = 12

BOOL = 13

NUM = 14

INT = 15

FLOAT = 16

LIST = 17

MATRIX = 18
