

61 Stream motion

This Stream motion function allows an external device to control a robot's motion by sending UDP packets through the Ethernet connection between the external device. The robot moves to the commanded position after a very short delay. In the following details, external device is regarded as PC. PC needs to calculate the interpolation point information considering velocity, acceleration/deceleration and jerk.

The following option provides this new functionality:

- J519 Stream motion

Only following robots are supported by this function currently.

- H755 LR-Mate 200iD
- H753 LR-Mate 200iD/7L
- H751 LR-Mate 200iD/7LC
- H628 M410iB/140H (Only series 7DC3)
- M-20iA/20(H844) V9.10P/12, V8.30P/43
- M710iC/50(H641) V9.10P/13, V8.30P/43
- Ordinary robots with six axes V9.10P/13, V8.30P/43. (expected)

The Stream motion function provides two new TP instructions:

IBGN start[*]

- * : Meaningless in this function. You can set from 1 to 16.
Robot wait command from PC by running this function.

IBGN end[*]

- * : Set the same number as set in IBGN start.
If PC finishes control, running line jumps to this command and program continues.
Program between IBGN start[*] and IBGN end[*] is not run.



WARNING

- 1 This function limits speed to less than a specific value, but if PC sends mistaken commands, robot may not move as expected, and serious personal injury could result. When using this function, you need to check correct command is output and thoroughly test movement before using.
If group motion is changed to DISABLE, you can check robot motion without robot actually moving.
You can check position and speed if you use DCS POSITION / SPEED CHECK FUNCTION.
- 2 This function requires PC to answer within communication interval which is normally 8ms.
Non real-time OS like Windows can't guarantee to answer in 8ms.
User should take countermeasure like using real-time OS.

Procedure

- 1 Strip a part of the shielded Category 5 or more twisted-pair cable (STP cable), which is NOT normal Ethernet cable, sheath to expose the metal shield and push the shield against the grounding plate with the clamping hardware as "Ethernet function OPERATOR'S MANUAL (B-82974)" of "D. CABLE CONNECTION".
- 2 Set PC and robot Ethernet port to communicate between PC and robot. About settings in robot, see "Ethernet function OPERATOR'S MANUAL (B-82974)" of "2 SETTING UP TCP/IP" of "2.1 OVERVIEW", "2.2 HARDWARE REQUIREMENTS AND INSTALLATION" and "2.4 SETTING UP TCP/IP". Please read and setup carefully as there are some detailed points that must be followed.
- 3 Set physical port number used by this function to \$STMO.\$PHYS_PORT. 1: CD38A, 2:CD38B
- 4 Teach TP program on robot controller. Bracket the section controlling robot from PC with "IBGN start[*]" / "IBGN end[*]" (* should be same number). IBGN start/end can be taught from pressing NEXT key some times -> F1 key [INST] -> "ASCII INTERFACE".

Protocol

Endian in communication data is Big endian.

- 1 Send following "Table 1 : Data output start packet" to robot port 60015 from PC by UDP. This packet can be sent always after robot is power ON.
- 2 By aforementioned step, following "Table 2 : Data format output from robot" are sent from robot to PC port, which sent aforementioned data (using same socket in other word), by UDP every constant interval. Communication interval is written in \$STMO.\$COM_INT[ms]. Communication interval is different by robot or setting but 8ms normally.
- 3 After bit 0 of status bit in "Table 2 : Data format output from robot" is changed to ON, by running IBGN start[*], set destination point with following "Table 3 : Command packet", and send the data to robot from PC by UDP to port 60015 after receiving Table2 packet.
Robot checks command packet by communication interval. If robot can't receive data in checking time, "MOTN-593 ST: Receiving interval over" [PAUSE] alarm occurs. This alarm doesn't happen at 1st packet.
- 4 See "Last data" in "Table 3 : Command packet" to finish control from PC and to finish packet, which is from PC, waiting status.
- 5 To stop output of "Table 2 : Data format output from robot", send "Table 4 : Data output stopping packet" to robot.

Rerference information

- 1 Starting program etc. can be done with HMI DEVICE COMMUNICATION function. See "Optional Function OPERATOR'S MANUAL" (B-83284EN-2).

Table 1 : Data output start packet (PC -> Robot)

Data type	Name	Description
4 byte Unsigned integer	Packet type	Set 0 : Data output start packet.
4 byte Unsigned integer	Version No.	Version No. of packet format and function. Set 1.

Table 2 : Data format output from robot (PC <- Robot)

Data type	Name	Description
4 byte Unsigned integer	Packet type	Fixed to 0 currently.
4 byte Unsigned integer	Version No.	Version No. of packet format and function. Set 1.
4 byte Unsigned integer	Sequence No.	Start from 1 and incremented 1 every sending. Next of 0xFFFFFFFF is 0. This is reset to 1 when Table1 : Data output start packet is received.
1 byte Unsigned integer	Status	Bit 0: 1:Command packet, which is described at Table 3, can be accepted (Changed to 1 by IBGN start command and changed to 0 by program pause/abort/Last data information from PC). Bit 1 : When command packet is received, this is changed to 1. This is changed to 0 at the same timing of Bit 0 changed to 0. Bit 2 1:SYSRDY ON 0:SYSRDY OFF (SYSRDY: System Ready) Bit 3 : 1 When robot is in motion. 0: when robot stopped. Bit 4-7: Unused
1 byte Unsigned integer	Read I/O type	DI:1, DO:2, RI:8, RO:9, SI:11, SO:12, WI:16, WO:17, UI:20, UO:21, WSI:26, WSO:27, F:35, M:36 If you don't read, 0 is set.
2 byte Unsigned integer	Read I/O index	Same with Reading I/O index in Table 3.
2 byte Unsigned integer	Read I/O mask	Same with Reading I/O mask in Table 3.
2 byte Unsigned integer	Read I/O value	This is the robot controller I/O value based on the setting of the read I/O index and read I/O mask specified by the motion command packet received by the controller in Table 3. Each bit represent either ON(1) or OFF(0) of the particular I/O port. This becomes 0 until packet described at Table 3 is received. Bit 0 (LSB) is I/O of "Read I/O index" and Bit 1 is I/O of "Read I/O index + 1" so bit n means I/O of "Read I/O index + n". I/O whose bit of "Read I/O mask" is 1 are read in continuous 16 points from

Data type	Name	Description
		<p>"Read I/O index" like following example.</p> <p>EXAMPLE.</p> <p>Read I/O index:0x8000, Read I/O mask:0x00f0, I/O value:0x8888</p> <p>-> Read I/O value:0x0080</p> <p>-> 0x8007:1 0x8006:0 0x8005:0 0x8004:0</p>
4 byte Unsigned integer	Time stamp	Time stamp when position data and motor current are recorded. Unit is ms. Resolution is 2ms. Next of 0xFFFFFFFF is 0.
4 byte float	X	Unit is mm.
4 byte float	Y	
4 byte float	Z	
4 byte float	W	Unit is degree.
4 byte float	P	
4 byte float	R	
4 byte float	Extended axis 1	Unit is mm if linear axis, degree if rotary axis.
4 byte float	Extended axis 2	
4 byte float	Extended axis 3	
4 byte float	J1	Unit is mm if linear axis, degree if rotary axis.
4 byte float	J2	
4 byte float	J3	
4 byte float	J4	
4 byte float	J5	
4 byte float	J6	
4 byte float	J7	
4 byte float	J8	
4 byte float	J9	
4 byte float	J1	Mortor current value [A].
4 byte float	J2	
4 byte float	J3	
4 byte float	J4	
4 byte float	J5	
4 byte float	J6	
4 byte float	J7	
4 byte float	J8	
4 byte float	J9	

Table 3 : Command packet (PC -> Robot)

Data type	Name	Description
4 byte Unsigned integer	Packet type	Set 1 : Command packet.
4 byte Unsigned integer	Version No.	Version No. of packet format and function. Set 1.
4 byte Unsigned integer	Sequence No.	Set same No. received from robot in communication interval.
1 byte Unsigned integer	Last data	<p>Normally 0, set 1 by PC if stopping control from PC. If this data is set to 1, robot finishes receive waiting status, which wait command packet from PC. After this data is set to 1, program running line jumps to IBGN END and program progress.</p> <p>You can send "Table 4 : Data output stopping packet" just after this data is set to 1. (But you don't have to send just after.)</p> <p>After sending with this data is 1, bit 0 and bit 1 in "Status" in "Table2 : Data format output from robot" may 1 in the next packet and set to 0 after the next packet.</p>
1 byte Unsigned integer	Reading I/O type	DI:1, DO:2, RI:8, RO:9, SI:11, SO:12, WI:16, WO:17, UI:20, UO:21, WSI:26, WSO:27, F:35, M:36 If you don't read, set 0.
2 byte	Reading I/O	I/O whose bit of "Reading I/O mask" is 1 are read in continuous 16 points

Unsigned integer	index	from "Reading I/O index". Bit 0 (LSB) is I/O of "Reading I/O index" and Bit 1 is I/O of "Reading I/O index + 1" so bit n means I/O of "Reading I/O index + n". ON is 1 and OFF is 0. EXAMPLE. Reading I/O index:0x8000, Reading I/O mask:0x00f0, I/O value:0x8888 -> Read I/O value:0x0080 -> 0x8007:1 0x8006:0 0x8005:0 0x8004:0	
2 byte Unsigned integer	Reading I/O mask		
1 byte Unsigned integer	Data style	This specifies position data in this packet is whether cartesian style or joint style. 0 : Cartesian data (XYZWPR) 1 : Joint DATA (J1, J2, J3, J4, J5, J6)	
1 byte Unsigned integer	Writing I/O type	DI:1, DO:2, RI:8, RO:9, SI:11, SO:12, WI:16, WO:17, UI:20, UO:21, WSI:26, WSO:27, F:35, M:36 Set 0 if you don't write.	
2 byte Unsigned integer	Writing I/O index	Writing corresponding "Writing I/O valule" bit whose "Writing I/O mask" bit is 1 in continuous 16 points start from "Writing I/O index", like following example. ON:1, OFF:0. EXAMPLE. Writing I/O index:0x8000, Writing I/O mask:0x00f0, Writing I/O value:0x8888 -> 8007:1 0x8006:0 0x8005:0 0x8004:0	
2 byte Unsigned integer	Writing I/O mask		
2 byte Unsigned integer	Writing I/O value		
2 byte	unused		
4 byte float	X or J1	0 : Cartesian style Unit is mm.	1 : Joint style Unit is degree.
4 byte float	Y or J2	Unit is degree.	
4 byte float	Z or J3		
4 byte float	W or J4		
4 byte float	P or J5		
4 byte float	R or J6	If linear axis, unit is mm. If rotary axis, unit is degree.	
4 byte float	Extended axis 1		
4 byte float	Extended axis 2		
4 byte float	Extended axis 3		

Table 4 : Data output stopping packet (PC -> Robot)

Data type	Name	Description
4 byte Unsigned integer	Packet type	Set 2 : Data output stopping packet.
4 byte Unsigned integer	Version No.	Version No. of packet format and function. Set 1.

Cartesian position data (X,Y,Z,W,P,R)

The position is TCP position with all 0 tool frame on world frame.

Tool frame and user frame are not checked when doing IBGN start[*].

61.1 Execution of Stream motion

- 1 During T1 or T2 mode or override isn't 100%, stream motion in isn't done with alarm. Alarm also happens and program is paused if override is degrade during command packet receiving
- 2 Don't send and receive 1 packet per 1ms in Ethernet port 1 and 2 in all. If you send and receive more frequency, packet may be discarded and alarm may happen.
- 3 If configuration in position data is changed during Stream motion when command packet is Cartesian style, "MOTN-156 PM: Config changed(L:%d)" alarm happens and paused.

61.2 System variables

Following system variables are saved in sysvars.sv.

System variable		default	Description
\$PM_GRP[]	\$ACC_TIME1	1	Acceleration/deceleration time in Stream motion. (1 st time constant). If 1, minimum acceleration/deceleration time.
	\$ACC_TIME2	1	Acceleration/deceleration time in Stream motion. (2 nd time constant). If 1, minimum acceleration/deceleration time.
	\$POS_ERR_LIM	16	Acceptable distance limit between previous interpolation position and current interpolation position during Stream motion. (Unit is mm)
	\$ROT_ERR_LIM	4.0	Acceptable orientation limit between previous interpolation position and current interpolation position during Stream motion. (Unit is degree)
	\$JVEL_RATIO	101	Acceptable joint axis between previous interpolation position and current interpolation position during Stream motion is executed. Percentage from maximum joint speed. (Unit is %)
\$STMO	\$PHYS_PORT	2	Set Ethernet port using by this function.
	\$COM_INT	8	Communication interval (Unit ms). (Can't be changed.)
	\$STCM_PRI	49	Stream motion tasks priority.
	\$STCR_PRI	49	1 is highest priority.
	\$STCS_PRI	10	255 is lowest priority (task doesn't get to run). If you set high priority, other function may influenced so least change is required.

61.3 Restrictions

- Constant Path (R663) needs to be enabled.
- Stream motion only supports single group motion streaming. If your system have multiple groups, the stream motion will work for group 1 only. That is, your TP program that runs IBGN START[]/IBGN END instruction should be a single group program and only group 1 is enabled.
- In program, which uses IBGN START/END, 30,000 or more can't be use as label of jump/label command.
- If IBGN END is executed before IBGN START instruction, the controller will post an alarm.
- The Stream Motion option can't be ordered with External Path Optimization (J829).
- Machine Locked Mode is NOT supported.
- The Stream Motion option can't be used with following options. (But you can load these options in the controller, as long as during the stream motion, these options are not active/enabled.)

continuous turn	J613	Torque Limit	J611
Touch sensor	J536	SoftFloat	J612
Weaving	J504	Coordinated Motion Package	J686
Line tracking	J512	Simultaneous Robot Link	J777
ARC sensor	J511	Arc related functions	
Root pass memorization	J532	AVC	J526