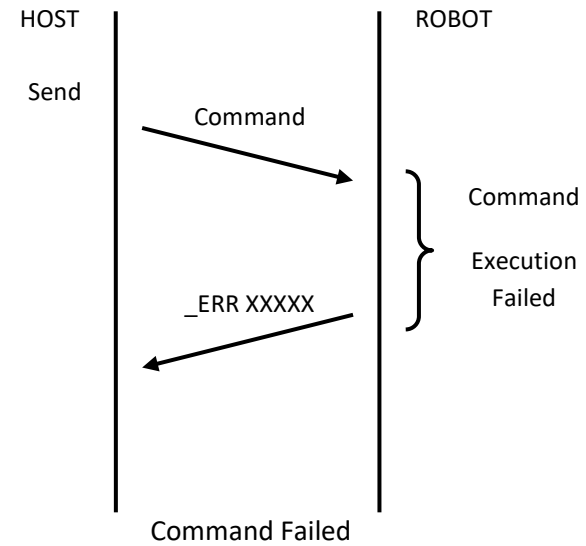
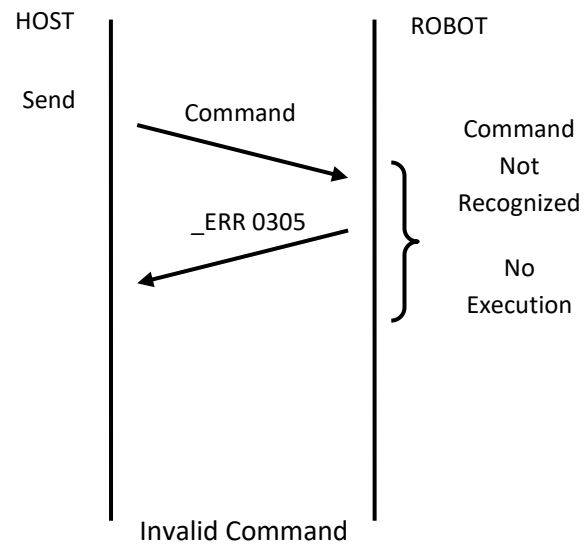
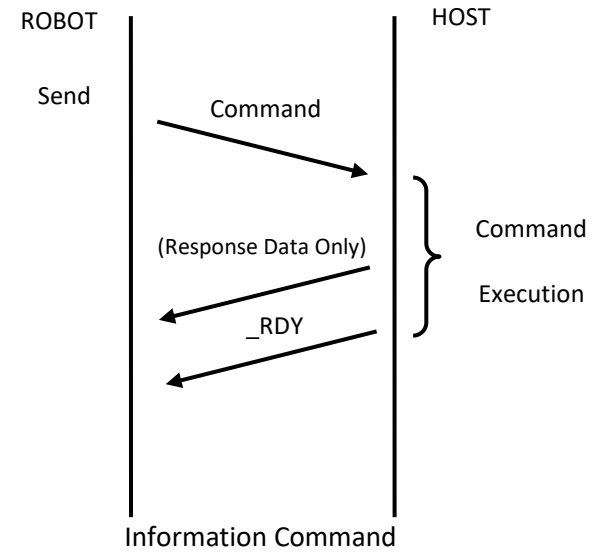
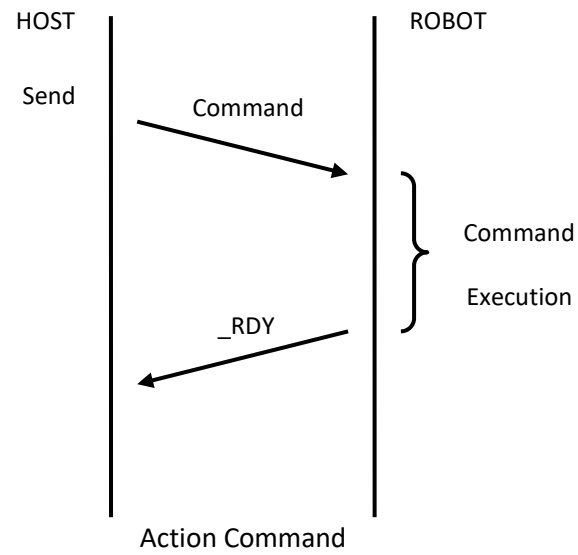


COMMUNICATION BETWEEN HOST COMPUTER AND ROBOT CONTROLLER

The following commands are available for a host computer to control the robot through the Ethernet or Serial ports.

Unique command prompts are available to enhance communication transactions between the host computer and the robot controller by highlighting command acknowledged, command not acknowledged, information response, error occurred, and command complete. These prompts can be configured individually to match host requirements. The default values are listed in the table to the right and typical host communication transactions are illustrated in the diagram below.

Transaction	Default Prompt	Custom Prompt
Command Acknowledged	_ACK	None
Command Not Acknowledged	_NAK	_ERR 305
Information Response	_RSP	None
Command Failed	_ERR	_ERR
Command Complete	_RDY	_RDY
Asynchronous Event	_EVT	_EVENT
Command Prompt	>	None
Process Command	<Carriage Return><Line Feed>	<Carriage Return><Line Feed>



HELLO

The HELLO command will perform no actions other than responding with a “hello” response back to the host. It is used to test communication between the host and robot.

Command	Parameter	Send Command
Hello	[none]	<CR><LF>
Example		Response
HELLO		Hello

HLLO

The HLLO command performs the same function as HELLO.

Command	Parameter	Send Command
Hllo	[none]	<CR><LF>
Example		Response
HLLO		Hello

The RQ BG command is used to request if the robot is currently executing a task or is idle.

Command	Robot <i>(Optional for single robot systems)</i>	Parameter	Send Command
RQ BG	robot1	[none]	<CR><LF>
Response Format (different than command input format)			
Robot	Response Header	Robot State	Error Code
Null (if no robot is specified)	BG	Y	0000 (no error)
Robot1		N	or
Robot2			10007 (error code if command failed)
Example		Response	
RQ BG (robot executing task)		BG Y 0000	
RQ BG (robot successfully finished task, sitting idle)		BG N 0000	
RQ BG (robot failed task with 10007 error)		BG N 10007	
Command Flow in custom “COMM FLOW BKG” mode		Command Flow in “COMM FLOW SEQ” mode (parallel tasks on unique task ID’s)	
<div>➔ PICK 1 (Action command)</div> <div>⬅ _RDY (immediate response, similar to acknowledgement)</div> <div>➔ RQ BG (request robot status)</div> <div>⬅ BG Y 0000 (robot executing a background task, pick command)</div> <div>⬅ _RDY</div> <div>➔ RQ BG</div> <div>⬅ BG Y 0000</div> <div>⬅ _RDY</div> <div>➔ RQ BG</div> <div>⬅ BG Y 0000</div> <div>⬅ _RDY</div> <div>➔ RQ BG</div> <div>⬅ BG Y 0000</div> <div>➔ RQ BG</div> <div>⬅ BG Y 0000</div> <div>➔ RQ BG</div> <div>⬅ BG N 0000 (robot successfully completed background task, pick command)</div>		<div>➔ 001 PICK 1 (Action command)</div> <div>➔ 002 RQ BG (request robot status)</div> <div>⬅ 002 BG Y 0000 (robot executing a background task, pick command)</div> <div>➔ 002 RQ BG</div> <div>⬅ 002 BG Y 0000</div> <div>➔ 002 RQ BG</div> <div>⬅ 002 BG Y 0000</div> <div>➔ 002 RQ BG</div> <div>⬅ 002 BG Y 0000</div> <div>➔ 002 RQ BG</div> <div>⬅ 002 BG Y 0000</div> <div>⬅ 001 _RDY</div> <div>➔ 002 RQ BG</div> <div>⬅ 002 BG N 0000 (robot successfully completed background task, pick command)</div>	

MOVEMENT COMMANDS

The following commands are available for a host computer to control the motion of the robot through the Ethernet or Serial ports.

GOTO

The GOTO command will move the specified robot arm to one of four possible positions at a station. The robot arm can be extended (EX) or retracted (RE) and the z height can be up (UP) at the wafer transfer height or down (DN) below the substrate handoff position for a specific substrate slot. These positions are determined by the station teach point coordinates and up/down offsets. All moves execute at the appropriate speeds determined by arm payload. Z position moves with the arm extended will execute using reduced speed values to minimize the speed during substrate transfer. Motion between retract and extend positions will pass through all intermediate via positions if configured. The GOTO command can also be used to move from one station's retract position to another station's retract position. The offset parameters (6 & 7) can be specified only if the final arm position is extended.

Command	Robot <i>(optional for single robot systems)</i>	Param 1 [station] <i>(optional if robot is currently at a valid station position)</i>	Param 2 [armPosition] <i>(optional)</i>	Param 3 [zPosition] <i>(optional)</i>	Param 4 [slot] <i>(optional)</i>	Param 5 [arm] <i>(optional)</i>	Param 6 [radialOffset] <i>(optional, offsets can only be applied to R EX position)</i> [Integer microns]	Param 7 [thetaOffset] <i>(optional, offsets can only be applied to R EX position)</i> [Integer millidegrees]	Send Command
GOTO	robot1	1 N 1 2 N 2 Etc...	R RE R EX	Z DN Z UP	SLOT 1 SLOT 2 SLOT 3 SLOT 4 Etc...	A ARM A B ARM B	RO 1234	TO 1234	<CR><LF>
Example									
GOTO robot1 N 1 R RE Z DN SLOT 1 ARM A or GOTO N 1 (in a single robot system)									

HOME

The HOME command initializes and moves the robot's arm(s) to a predefined reference position. Each axis (R,Z,T) can be referenced in order individually or the entire home sequence can be executed by specifying "ALL". When retracting the arm (home R), the robot will attempt to retract through a known safe path to avoid obstructions. If a path is found within a close proximity of the robot's current location, the robot will move to the path and retract along the path through all associated via positions. If no path is found, the robot will retract on a direct radial path. All moves are executed using the "home" speed set.

Command	Robot <i>(optional for single robot systems)</i>	Parameter 1 [axis]	Send Command
HOME	robot1	R T Z ALL	<CR><LF>
Example			
HOME robot1 ALL or HOME ALL (in a single robot system)			

PICK

The PICK command is used to pick up and remove a wafer from a station. The PICK command combined with a PLACE command will allow the host to move substrates from one station to another.

Command	Robot (optional for single robot systems)	Param 1 [station] (optional if robot is currently at a valid station position)	Param 2 [slot] (optional, defaults to "slot 1" if not sent)	Param 3 [arm] (optional, defaults to A if not sent)	Param 4 [end effector] (optional, defaults to ALL EE's if not sent)	Param 5 [radialOffset] (optional, defaults to zero if not sent) [Integer microns]	Param 6 [thetaOffset] (optional, defaults to zero if not sent) [Integer millidegrees]	Param 7 [Stop or Start Point] (optional, Stops or starts transfer at teach point)	Param 8 [Force APS] (optional, Force APS calculation)	Send Command
PICK	robot1	1 2 3 Etc...	SLOT 1 SLOT 2 SLOT 3 SLOT 4 Etc...	A ARM A B ARM B	PAN L PAN R	RO 1234	TO 1234	ENRT NR (end at teach point) STRT NR (Start at teach point)	APS	<CR><LF>
Example										
PICK robot1 1 SLOT 1 ARM A RO 1234 TO 1234 ENRT NR										

PLACE

The PLACE command is used to place a wafer into a station. The PLACE command combined with a PICK command will allow the host to move substrates from one station to another.

Command	Robot (optional for single robot systems)	Param 1 [station] (optional if robot is currently at a valid station position)	Param 2 [slot] (optional, defaults to "slot 1" if not sent)	Param 3 [arm] (optional, defaults to A if not sent)	Param 4 [end effector] (optional, defaults to ALL EE's if not sent)	Param 5 [radialOffset] (optional, defaults to zero if not sent) [Integer microns]	Param 6 [thetaOffset] (optional, defaults to zero if not sent) [Integer millidegrees]	Param 7 [Stop or Start Point] (optional, Stops or starts transfer at teach point)	Param 8 [Force APS] (optional, Force APS calculation)	Send Command
PLACE	robot1	1 2 3 Etc...	SLOT 1 SLOT 2 SLOT 3 SLOT 4 Etc...	A ARM A B ARM B	PAN L PAN R	RO 1234	TO 1234	ENRT NR (end at teach point) STRT NR (Start at teach point)	APS	<CR><LF>
Example										
PLACE robot1 1 SLOT 1 ARM A RO 1234 TO 1234 ENRT NR										

PAYLOAD COMMANDS

The following commands are available for a host computer to evaluate and set the payload status of the robot through the Ethernet or Serial ports.

CHECK LOAD

The CHECK LOAD command is used to check the actual payload state of an end effector using an external wafer presence sensor connected to the robot's digital inputs. The wafer presence sensors are typically located along the extension path to each station. To validate the payload on the end effector the robot must move to the retract position of the specified station, extend the arm to the specified radial position, and check the payload sensor state. The sensor state will be used to set the logical payload state in the object. The interlock option will temporarily enable or disable the specified interlock / handshake for the duration of the command execution. The interlock will resume normal operation when the command is complete.

Command	Robot <i>(optional for single robot systems)</i>	Parameter 1 [station] <i>(optional, searches for closest station with sensor if not sent)</i>	Parameter 2 [arm] <i>(optional, defaults to A if not sent)</i>	Parameter 3 [interlock option] <i>(optional, defaults to Enabled if not sent)</i>	Send Command
CHECK LOAD	robot1	1 2 3 Etc...	A ARM A B ARM B	INTLCK EX_ENABLE DIS INTLCK EX_ENABLE ENB	<CR><LF>
Example					
CHECK LOAD robot1 1 ARM A INTLCK EX_ENABLE DIS or CHECK LOAD 1 ARM A (in a single robot system)					

RQ LOAD

The RQ LOAD command is used to check the logical payload state of an end effector. The response of this command will be determined by the robot.arm load state.

Command	Robot <i>(optional for single robot systems)</i>	Command	Parameter 1 [arm] <i>(optional, defaults to Arm A if not provided)</i>	Parameter 2 [endEffector] <i>(optional, defaults to PAN L if not sent, not used for single EE/Arm robots)</i>	Send Command
RQ	robot1	LOAD	A ARM A B ARM B	PAN L PAN R	<CR><LF>
Example			Response		
RQ LOAD (in a single robot system) or RQ robot1 LOAD ARM A or RQ robot1 LOAD ARM A PAN L			LOAD A ON LOAD A PAN L ON		

SET LOAD

The SET LOAD command allows the host to set the current payload tracking state from the robot controller. This is a recommended step for the host during system power up and system error recovery.

Command	Robot <i>(optional for single robot systems)</i>	Command	Parameter 1 [arm] <i>(optional, defaults to arm A if not provided)</i>	Parameter 2 [endEffector] <i>(optional, defaults to PAN A if not sent)</i>	Parameter 3 [Load State]	Send Command
SET	robot1	LOAD	A ARM A B ARM B	PAN L PAN R	ON Y OFF N	<CR><LF>
Example						
SET robot1 LOAD ARM A PAN L ON or SET LOAD ON (in a single robot single arm system)						

RQ POSITION ABS

The RQ POS ABS command is used to request the specified robot and arm's current position in absolute cylindrical coordinate reference frame.

- These parameters can be requested individually or in multiples in any order.
- All values can be requested by specifying "ALL".
- The parameters received by the robot controller can be in any order, but the response order shall follow the table order outlined below.
- If an individual parameter is omitted in the command, the corresponding response will not include the value for that parameter.
- If no arm parameter is sent, the value will default to A for execution, but no arm value ("A" or "B") will be included in the response string. If the arm is included in the command, the arm value ("A" or "B") will be included in the response string.

Command	Robot <i>(optional for single robot systems)</i>	Command	Parameter 1 [arm] <i>(optional, defaults to A if not sent)</i>	Parameter 2 [Radial Coordinate] <i>(optional)</i>	Parameter 3 [Theta Coordinate] <i>(optional)</i>	Parameter 4 [Z Coordinate] <i>(optional)</i>	Send Command
RQ	robot1	POS ABS	A ARM A B ARM B	R	T	Z	<CR><LF>
Individual Responses	Null <i>(if no robot is specified)</i> robot1 robot2		Null <i>(if no arm is specified)</i> A B	0001234 (7 char)	360000 (6 char)	150000 (6 char)	
Example				Response			
RQ POS ABS robot1 1 ARM A R T Z				POS ABS robot1 A 0001234 360000 150000			
RQ POS ABS Z T R (reversed order from above)				POS ABS 0001234 360000 150000 (same order as above)			
RQ POS ABS B ALL				POS ABS B 0001234 360000 150000			
RQ POS ABS ALL				POS ABS 0001234 360000 150000			

RQ POSITION STN

The RQ POS STN command is used to determine if the robot is at a known station move position or home position.

- These parameters can be requested individually or in multiples in any order.
- All values can be requested by specifying "all".
- The parameters received by the robot controller can be in any order, but the response order shall follow the table order outlined below.
- If an individual parameter is omitted in the command, the corresponding response will not include the value for that parameter.
- If no arm parameter is sent, the value will default to A for execution, but no arm value ("A" or "B") will be included in the response string.

If the arm is included in the command, the arm value ("A" or "B") will be included in the response string.

Command	Robot <i>(optional for single robot systems)</i>	Command	Parameter 1 [arm] <i>(optional, defaults to A if not sent)</i>	Parameter 2 [Radial Coord.] <i>(optional)</i>	Parameter 3 [Theta Coord.] <i>(optional)</i>	Parameter 4 [Z Coord.] <i>(optional)</i>	Parameter 5 [Down Offset] <i>(optional) (No Action)</i>	Parameter 6 [Slot Count] <i>(optional) (No Action)</i>	Parameter 7 [Slot Pitch] <i>(optional) (No Action)</i>	Parameter 8 [Slot Pitch] <i>(optional)</i>	Send Command
RQ	robot1	POS STN	A, ARM A B, ARM B	R	T	Z	W	WA	WB	SLOT	<CR><LF>
Response Format <i>(different than command input format)</i>											
Header	Robot	Arm	Radial Position	Station	Slot	Z Position	Theta Position				
POS STN	Null <i>(if no robot is specified)</i>	Null <i>(if no arm is specified)</i>	RE	00 <i>(not at station)</i>	0000	UP	360000				
			EX	01	0001	DN	<i>(6 char theta position)</i>				
	Robot1	A	HM	02	<i>(4 char slot number)</i>	HM	or				
	Robot2	B	-- <i>(2 char)</i>	03	Or	--	HM				
				-- <i>(2 char)</i>	-- <i>(2 char unknown)</i>	-- <i>(2 char)</i>	-- <i>(2 char)</i>				
Example				Response							
RQ POS STN robot1 ARM A ALL				POS STN robot1 A RE 01 0001 DN 360000							
RQ POS STN ALL				POS STN RE 01 0001 DN 360000							
RQ POS STN ALL(robot at home position)				POS STN HM 00 0000 HM HM							
RQ POS STN ARM A ALL (Position unknown)				POS STN A -- -- -- --							
RQ POS STN ARM A R Z				POS STN A EX UP							

RQ WAF_CEN DATA

The RQ WAF_CEN DATA command is used to request the calculated leading and trailing wafer edge transitions for each sensor and the calculated correction offset immediately following a robot transfer with APS

Command				Robot <i>(optional for single robot systems)</i>				Command				Send Command					
RQ				robot1				WAF_CEN DATA				<CR><LF>					
Example																	
RQ robot1 WAF_CEN DATA or RQ WAF_CEN DATA <i>(in a single robot system)</i>																	
Response format <i>(different than command input format - 6 characters)</i>																	
Header	Right Sensor Identifier	Right Sensor Leading Edge Radial Value [microns]	Right Sensor Leading Edge Theta Value [millideg]	Right Sensor Trailing Edge Radial Value [microns]	Right Sensor Trailing Edge Theta Value [millideg]	Left Sensor Identifier	Left Sensor Leading Edge Radial Value [microns]	Left Sensor Leading Edge Theta Value [millideg]	Left Sensor Trailing Edge Radial Value [microns]	Left Sensor Trailing Edge Theta Value [millideg]	Center Sensor Identifier	Center Sensor Leading Edge Radial Value [microns]	Center Sensor Leading Edge Theta Value [millideg]	Center Sensor Trailing Edge Radial Value [microns]	Center Sensor Trailing Edge Theta Value [millideg]	Correction Offset Radial Value [microns]	Correction Offset Theta Value [millideg]
WAF_CEN	RT	000000	000000	000000	000000	LFT	000000	000000	000000	000000	CEN	000000	000000	000000	000000	000000	000000
WAF_CEN DATA RIGHT_SENS 575856 335789 386957 335787 LEFT_SENS 582726 335785 379129 335790 OFFSET 000038 000000 -00001 000038																	

CUSTOM2 PARSER ERROR CODE LIST

The following is a comprehensive list of ERR XXXXX codes that can be returned when a command is not recognized, or execution of the command failed.

Error Number	Error Name	Native Error
210	NOT_AT_STATION	ERR_NOTATSTATION
220	NOT_RETRACTED	ERR_NOTRETRACTED
221	NO_ACTIVE_ARM	ERR_NOACTIVEARM
233	HANDSHAKE	ERR_HANDSHAKE
305	INVALID_COMMAND	ERR_SYNTAXERROR
309	NOT_SUPPORTED	ERR_NOTIMPLEMENTED
352	NO_MEMORY	ERR_NOMEMORY
443	INVALID_STATION	ERR_STNINVALID
450	INVALID_STATION_POSITION	ERR_STNOUTOFRANGE
451	APS_DEVIATION_LIMIT	ERR_APSCORROVERLIMIT
509	Z_LESS	ERR_ZLESS
510	W_LESS	ERR_WLESS
553	S_LESS	ERR_SLESS
527	MCC_COM	ERR_COMMUNDEFINEDERROR ERR_COMMNOPARSEABLEMESSAGE ERR_COMMWRONGPARM ERR_COMMINTERNALERROR ERR_COMMMEMORYLEAK ERR_COMMSSENDPROBLEM ERR_COMMNOFREEOUTPUTSLOT ERR_PARAM_ARRAYNULL ERR_PARAM_SIZE mismatch ERR_PARAM_NOTIMPEMENTED ERR_SYSTEMCALLERR

Error Number	Error Name	Native Error
528	MCC_QUEUE	ERR_BUFFFULL
558	UNKNOWN_ERROR	ERR_UNKNOWN ERR_NULLCALLPARAM
602	BUSY	ERR_RESOURCEBUSY ERR_RESEACTIVE
603	HALTED	ERR_CMDABORTED
607	NO_MCC	ERR_COMMNOTCONNECTED ERR_COMMTIMEOUT ERR_COMMCOMMUNICATIONSTOPPED
608	HALTING	ERR_TRJABORTED
658	NOT_ALLOWED	ERR_HWOFFLINE
705	PAYLOAD_MISSING	ERR_PAYLOADMISSING
706	PAYLOAD_PRESENT	ERR_PAYLOADPRESENT
770	APS_INVALID_SENSOR	ERR_APS_SENSORPOSINVALID
771	APS_INVALID_DATA	
772	APS_BROKEN_WAFER	ERR_APSWAFERBREAKAGE
773	APS_LARGE_BURST	ERR_APS_SENSORNOISE
774	APS_INVALID_BURSTS	ERR_APS_EDGEVALUEINVALID
775	APS_DISABLED	ERR_POSCPTRNOTACTIVE ERR_APS_DISABLED ERR_APS_NOTCALIBRATED
776	APS_MISSING_DATA	ERR_APS_CPTRPOSMISSING ERR_APS_EDGECOUNTINVALID
777	APS_CALCULATE	ERR_APS_CORRITEFAIL ERR_APS_UNKNOWN
778	APS_NO_SENSOR	

Error Number	Error Name	Native Error
821	OBJECT_NOT_FOUND	ERR_COREOBJNOTEXISTS ERR_BLUEOBJNOTEXISTS ERR_CONFOBJNOTFOUND
822	INVALID_OBJECT	ERR_CONFIGINVALID ERR_DUPOBJECTNAME
823	BAD_TYPE	ERR_COREOBJDIFFTYPE
830	NOT_CONFIGURED	ERR_NOTCONFIGURED
832	READ_ONLY	ERR_READONLY
838	INVALID_NAME	ERR_PARM_NOTEXISTS
840	INVALID_TYPE	ERR_PARAM_TYPEMISMATCH ERR_PARAM_NOTARRAY
844	INVALID_PARAM	ERR_APS_WAFERDIAMINVALID ERR_PARAM_INVALID ERR_PARAM_IDMINUS1 ERR_PARAM_ENUMINVALID ERR_PARAM_VALUENAN
951	EVENT_NOT_FOUND	ERR_EVENT_NOTFOUND
952	UNKNOWN_EVENT	ERR_EVENT_NOTSUPPORTED
1101	WORK_SPACE	ERR_TRJOUTOFSAFE
1805	CDM_CONTROL	ERR_CTRLLOCKED
2003	TIMEOUT	ERR_SWTIMEOUT
3000	TRACE_ACTIVE	ERR_TRACEACTIVE
3001	ALREADY_CONFIGURED	ERR_ALREADYCONFIGURED
4007	LOG_ERROR	ERR_LOGERROR
6010	OVERTEMP	ERR_MTOVERTEMP ERR_OVERTEMP
6011	PRESSURE	ERR_PRESSURE
10007	SETTLE_TIMEOUT	ERR_SETTTIMEOUT
10009	HARDTRACK	ERR_HARDTRACK

Error Number	Error Name	Native Error
10012	MOTOR_CONFIG	ERR_HWCONFIGINVALID
10013	NOT_REFERENCED	ERR_MTRPWROFF ERR_SERVOOFF
10014	REFERENCING	
10015	MOVING	ERR_MOVING
10016	TRAJECTORY	ERR_TRJGENERR
10019	ENCODER	ERR_ENCFAULT
10021	POWER	ERR_UNDERVOLTAGE ERR_OVERVOLTAGE ERR_POWERFAULT
10022	OVERTRAVEL_LIMIT	ERR_TRJOUTOFRANGE
10024	MCC_TIMEOUT	ERR_HWTIMEOUT
10029	EMERGENCY_STOP	ERR_ESTOP ERR_INTERLOCK ERR_SAFETYFAULT
10030	OVERCURRENT	ERR_OVERCURRENT ERR_SHORT
10061	TRJ_SINGULARITY	ERR_TRJSINGULARITY
10067	BAD_SYNC_STATE	ERR_HWOffline