ESP301

Integrated 3-Axis Motion Controller/Driver

Commande Interface Manual

For Motion, Think Newport™
Table of Contents

1.0 Introduction .................................................................................................................. 1
  1.1 Purpose ..................................................................................................................... 1
  1.2 Overview .................................................................................................................. 1

2.0 Command Interface....................................................................................................... 2
  2.1 Constructor ............................................................................................................... 2
  2.2 Functions .................................................................................................................. 2
    2.2.1 General ............................................................................................................... 2
      ♦ CloseInstrument ................................................................................................. 2
      ♦ GetDevices .......................................................................................................... 2
      ♦ OpenInstrument ................................................................................................. 2
      ♦ WriteToInstrument ............................................................................................. 3
    2.2.2 Commands ........................................................................................................ 3
      ♦ AB .......................................................................................................................... 3
      ♦ AC_Get ................................................................................................................... 3
      ♦ AC_Set ................................................................................................................... 3
      ♦ AE_Get ................................................................................................................... 4
      ♦ AE_Set ................................................................................................................... 4
      ♦ AF_Get ................................................................................................................... 4
      ♦ AF_Set ................................................................................................................... 4
      ♦ AG_Get ................................................................................................................... 5
      ♦ AG_Set ................................................................................................................... 5
      ♦ AP ........................................................................................................................... 5
      ♦ AU_Get ................................................................................................................... 5
      ♦ AU_Set ................................................................................................................... 6
      ♦ BA_Get ................................................................................................................... 6
      ♦ BA_Set ................................................................................................................... 6
      ♦ BG_Get ................................................................................................................... 6
      ♦ BG_Set ................................................................................................................... 7
      ♦ BK_Get ................................................................................................................... 7
      ♦ BK_Set ................................................................................................................... 7
      ♦ BL_Get ................................................................................................................... 7
      ♦ BL_Set ................................................................................................................... 8
      ♦ BM_Get ................................................................................................................... 8
      ♦ BM_Set ................................................................................................................... 8
      ♦ BN_Get ................................................................................................................... 8
      ♦ BN_Set ................................................................................................................... 9
      ♦ BO_Get ................................................................................................................... 9
ESP301
Integrated 3-Axis Motion Controller/Driver

- BO_Set................................................................................................................... 9
- BP_Get.................................................................................................................... 9
- BP_Set.................................................................................................................. 10
- BQ_Get.................................................................................................................. 10
- BQ_Set.................................................................................................................. 10
- CL_Get................................................................................................................... 10
- CL_Set.................................................................................................................. 11
- CO_Get.................................................................................................................. 11
- CO_Set.................................................................................................................. 11
- DB_Get.................................................................................................................. 11
- DB_Set.................................................................................................................. 12
- DC......................................................................................................................... 12
- DD......................................................................................................................... 12
- DE......................................................................................................................... 12
- DF......................................................................................................................... 13
- DG......................................................................................................................... 13
- DH_Get.................................................................................................................. 13
- DH_Set.................................................................................................................. 13
- DL......................................................................................................................... 14
- DO_Get.................................................................................................................. 14
- DO_Set.................................................................................................................. 14
- DP......................................................................................................................... 14
- DV......................................................................................................................... 15
- EO_Get.................................................................................................................. 15
- EO_Set.................................................................................................................. 15
- EP......................................................................................................................... 15
- EX......................................................................................................................... 16
- FE_Get.................................................................................................................. 16
- FE_Set.................................................................................................................. 16
- FP_Get.................................................................................................................. 16
- FP_Set.................................................................................................................. 17
- FR_Get.................................................................................................................. 17
- FR_Set.................................................................................................................. 17
- GR_Get.................................................................................................................. 17
- GR_Set.................................................................................................................. 18
- HA_Get.................................................................................................................. 18
- HA_Set.................................................................................................................. 18
- HB......................................................................................................................... 18
- HC_Get.................................................................................................................. 19
- HC_Set.................................................................................................................. 19
- HD_Get.................................................................................................................. 19
- HD_Set.................................................................................................................. 19
- HE_Get.................................................................................................................. 20
- HE_Set.................................................................................................................. 20
<table>
<thead>
<tr>
<th>Function</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>HF_Get</td>
<td>20</td>
</tr>
<tr>
<td>HF_Set</td>
<td>20</td>
</tr>
<tr>
<td>HJ_Get</td>
<td>21</td>
</tr>
<tr>
<td>HJ_Set</td>
<td>21</td>
</tr>
<tr>
<td>HL_Get</td>
<td>21</td>
</tr>
<tr>
<td>HL_Set</td>
<td>21</td>
</tr>
<tr>
<td>HN_Get</td>
<td>22</td>
</tr>
<tr>
<td>HN_Set</td>
<td>22</td>
</tr>
<tr>
<td>HO_Get</td>
<td>22</td>
</tr>
<tr>
<td>HO_Set</td>
<td>22</td>
</tr>
<tr>
<td>HP</td>
<td>23</td>
</tr>
<tr>
<td>HQ_Get</td>
<td>23</td>
</tr>
<tr>
<td>HQ_Set</td>
<td>23</td>
</tr>
<tr>
<td>HS_Get</td>
<td>23</td>
</tr>
<tr>
<td>HS_Set</td>
<td>24</td>
</tr>
<tr>
<td>HV_Get</td>
<td>24</td>
</tr>
<tr>
<td>HV_Set</td>
<td>24</td>
</tr>
<tr>
<td>HW</td>
<td>24</td>
</tr>
<tr>
<td>HX</td>
<td>25</td>
</tr>
<tr>
<td>HZ</td>
<td>25</td>
</tr>
<tr>
<td>ID</td>
<td>25</td>
</tr>
<tr>
<td>JH_Get</td>
<td>25</td>
</tr>
<tr>
<td>JH_Set</td>
<td>26</td>
</tr>
<tr>
<td>JK_Get</td>
<td>26</td>
</tr>
<tr>
<td>JK_Set</td>
<td>26</td>
</tr>
<tr>
<td>JL</td>
<td>26</td>
</tr>
<tr>
<td>JW_Get</td>
<td>27</td>
</tr>
<tr>
<td>JW_Set</td>
<td>27</td>
</tr>
<tr>
<td>KD_Get</td>
<td>27</td>
</tr>
<tr>
<td>KD_Set</td>
<td>27</td>
</tr>
<tr>
<td>KI_Get</td>
<td>28</td>
</tr>
<tr>
<td>KI_Set</td>
<td>28</td>
</tr>
<tr>
<td>KP_Get</td>
<td>28</td>
</tr>
<tr>
<td>KP_Set</td>
<td>28</td>
</tr>
<tr>
<td>KS_Get</td>
<td>29</td>
</tr>
<tr>
<td>KS_Set</td>
<td>29</td>
</tr>
<tr>
<td>LC_Get</td>
<td>29</td>
</tr>
<tr>
<td>LC_Set</td>
<td>29</td>
</tr>
<tr>
<td>LP</td>
<td>30</td>
</tr>
<tr>
<td>MD</td>
<td>30</td>
</tr>
<tr>
<td>MF_Get</td>
<td>30</td>
</tr>
<tr>
<td>MF_Set</td>
<td>30</td>
</tr>
<tr>
<td>MO_Get</td>
<td>31</td>
</tr>
<tr>
<td>MO_Set</td>
<td>31</td>
</tr>
<tr>
<td>Function</td>
<td>Page</td>
</tr>
<tr>
<td>----------</td>
<td>------</td>
</tr>
<tr>
<td>MT_Get</td>
<td>31</td>
</tr>
<tr>
<td>MT_Set</td>
<td>31</td>
</tr>
<tr>
<td>MV_Get</td>
<td>32</td>
</tr>
<tr>
<td>MV_Set</td>
<td>32</td>
</tr>
<tr>
<td>MZ_Get</td>
<td>32</td>
</tr>
<tr>
<td>MZ_Set</td>
<td>32</td>
</tr>
<tr>
<td>OH_Get</td>
<td>33</td>
</tr>
<tr>
<td>OH_Set</td>
<td>33</td>
</tr>
<tr>
<td>OL_Get</td>
<td>33</td>
</tr>
<tr>
<td>OL_Set</td>
<td>33</td>
</tr>
<tr>
<td>OM_Get</td>
<td>34</td>
</tr>
<tr>
<td>OM_Set</td>
<td>34</td>
</tr>
<tr>
<td>OR</td>
<td>34</td>
</tr>
<tr>
<td>PA_Get</td>
<td>34</td>
</tr>
<tr>
<td>PA_Set</td>
<td>35</td>
</tr>
<tr>
<td>PH</td>
<td>35</td>
</tr>
<tr>
<td>PR</td>
<td>35</td>
</tr>
<tr>
<td>QD</td>
<td>35</td>
</tr>
<tr>
<td>QG_Get</td>
<td>36</td>
</tr>
<tr>
<td>QG_Set</td>
<td>36</td>
</tr>
<tr>
<td>QI_Get</td>
<td>36</td>
</tr>
<tr>
<td>QI_Set</td>
<td>36</td>
</tr>
<tr>
<td>QM_Get</td>
<td>37</td>
</tr>
<tr>
<td>QM_Set</td>
<td>37</td>
</tr>
<tr>
<td>QP</td>
<td>37</td>
</tr>
<tr>
<td>QR_Get</td>
<td>37</td>
</tr>
<tr>
<td>QR_Set</td>
<td>37</td>
</tr>
<tr>
<td>QS_Get</td>
<td>38</td>
</tr>
<tr>
<td>QS_Set</td>
<td>38</td>
</tr>
<tr>
<td>QT_Get</td>
<td>38</td>
</tr>
<tr>
<td>QT_Set</td>
<td>38</td>
</tr>
<tr>
<td>QV_Get</td>
<td>39</td>
</tr>
<tr>
<td>QV_Set</td>
<td>39</td>
</tr>
<tr>
<td>RQ</td>
<td>39</td>
</tr>
<tr>
<td>RS</td>
<td>40</td>
</tr>
<tr>
<td>SA_Get</td>
<td>40</td>
</tr>
<tr>
<td>SA_Set</td>
<td>40</td>
</tr>
<tr>
<td>SB_Get</td>
<td>40</td>
</tr>
<tr>
<td>SB_Set</td>
<td>41</td>
</tr>
<tr>
<td>SH_Get</td>
<td>41</td>
</tr>
<tr>
<td>SH_Set</td>
<td>41</td>
</tr>
<tr>
<td>SI_Get</td>
<td>41</td>
</tr>
<tr>
<td>SI_Set</td>
<td>42</td>
</tr>
<tr>
<td>SK_Get</td>
<td>42</td>
</tr>
</tbody>
</table>
- SK_Set................................................................. 42
- SL_Get................................................................. 42
- SL_Set ............................................................... 43
- SM ....................................................................... 43
- SN_Get ............................................................... 43
- SN_Set ............................................................... 43
- SR_Get ............................................................... 44
- SR_Set ............................................................... 44
- SS_Get ............................................................... 44
- SS_Set ............................................................... 44
- ST ........................................................................ 45
- SU_Get ............................................................... 45
- SU_Set ............................................................... 45
- TB ........................................................................ 45
- TE ........................................................................ 46
- TJ_Get ............................................................... 46
- TJ_Set ............................................................... 46
- TP ........................................................................ 46
- TS ........................................................................ 47
- TV ........................................................................ 47
- TX ........................................................................ 47
- UF ........................................................................ 47
- UF ........................................................................ 47
- UF ........................................................................ 48
- UH ........................................................................ 48
- UL ........................................................................ 48
- VA_Get ............................................................... 48
- VA_Set ............................................................... 49
- VB_Get ............................................................... 49
- VB_Set ............................................................... 49
- VE ........................................................................ 49
- VF_Get ............................................................... 50
- VF_Set ............................................................... 50
- VU_Get ............................................................... 50
- VU_Set ............................................................... 50
- WP ........................................................................ 51
- WS ........................................................................ 51
- WT ........................................................................ 51
- XM ........................................................................ 51
- XX ........................................................................ 52
- ZA_Get ............................................................... 52
- ZA_Set ............................................................... 52
- ZB_Get ............................................................... 52
- ZB_Set ............................................................... 53
- ZE_Get ............................................................... 53
3.0 Python Example ................................................................. 56

Service Form .............................................................................. 59
1.0 Introduction

1.1 Purpose
The purpose of this document is to provide the method Syntax of each command to communicate with the ESP301 device.

1.2 Overview
The Command Interface is the wrapper class that maintains a list of ESP301 instruments. It exposes methods to communicate with any ESP301 device.

NOTE
Each function name is defined with the command code “AA”.
For each command function, refer to the ESP301 programmer’s manual.
2.0 Command Interface

2.1 Constructor

ESP301()

The constructor is used to create an instance of the ESP301 device.

2.2 Functions

2.2.1 General

✦ CloseInstrument

Syntax

int CloseInstrument()
return: 0 = successful else failure

Description

This function allows closing communication with the selected device.

✦ GetDevices

Syntax

string[] GetDevices()
return: list of strings that contains the accessible COM ports

Description

This function allows opening communication with the selected device.

✦ OpenInstrument

Syntax

int OpenInstrument(string strCOMPort, int baudrate)
string strCOMPort: COM port
int baudrate: baud rate
return: 0 = successful else failure

Description

This function allows opening communication with the selected device.
- **WriteToInstrument**

  **Syntax**
  ```c
  int WriteToInstrument(string command, ref string response, int stage)
  command: Instrument command
  response: Response of the command
  stage: Instrument Stage
  return:
  ```

  **Description**
  This overridden function queries or writes the command given by the user to the instrument.

2.2.2 **Commands**

- **AB**

  **Syntax**
  ```c
  int AB(out string errstring)
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous AB Set command which is used to Abort Motion.

- **AC_Get**

  **Syntax**
  ```c
  int AC_Get(int axisNumber, out double acceleration, out string errstring)
  axisNumber: axisNumber
  acceleration: acceleration
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous AC Get command which is used to Get acceleration.

- **AC_Set**

  **Syntax**
  ```c
  int AC_Set(int axisNumber, double acceleration, out string errstring)
  axisNumber: axisNumber
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous AC Set command which is used to Set acceleration.
AE_Get

Syntax
int AE_Get(int axisNumber, out double deceleration, out string errstring)
axisNumber: axisNumber
deceleration: deceleration
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous AE Get command which is used to Get e-stop deceleration.

AE_Set

Syntax
int AE_Set(int axisNumber, double deceleration, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous AE Set command which is used to Set e-stop deceleration.

AF_Get

Syntax
int AF_Get(int axisNumber, out double accelerationFeedForwardGainFactor, out string errstring)
axisNumber: axisNumber
accelerationFeedForwardGainFactor: accelerationFeedForwardGainFactor
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous AF Get command which is used to Get acceleration feed-forward gain.

AF_Set

Syntax
int AF_Set(int axisNumber, double accelerationFeedForwardGainFactor, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous AF Set command which is used to Set acceleration feed-forward gain.
♦ **AG_Get**

**Syntax**
```c
int AG_Get(int axisNumber, out double acceleration, out string errstring)
```
- `axisNumber`: axisNumber
- `acceleration`: acceleration
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous AG Get command which is used to Get deceleration.

♦ **AG_Set**

**Syntax**
```c
int AG_Set(int axisNumber, double acceleration, out string errstring)
```
- `axisNumber`: axisNumber
- `acceleration`: acceleration
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous AG Set command which is used to Set deceleration.

♦ **AP**

**Syntax**
```c
int AP(out string errstring)
```
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous AP Set command which is used to Abort program.

♦ **AU_Get**

**Syntax**
```c
int AU_Get(int axisNumber, out double acceleration, out string errstring)
```
- `axisNumber`: axisNumber
- `acceleration`: acceleration
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous AU Get command which is used to Get maximum acceleration and deceleration.
- **AU_Set**

  **Syntax**
  ```c
  int AU_Set(int axisNumber, double acceleration, out string errstring)
  ```
  axisNumber: axisNumber[InCommentDoc]
  errString: The failure reason
  return: 0 in success and -1 on failure

  **Description**
  This function is used to process synchronous AU Set command which is used to Set maximum acceleration and deceleration.

- **BA_Get**

  **Syntax**
  ```c
  int BA_Get(int axisNumber, out double backlashCompensation, out string errstring)
  ```
  axisNumber: axisNumber
  backlashCompensation: backlashCompensation
  errString: The failure reason
  return: 0 in success and -1 on failure

  **Description**
  This function is used to process synchronous BA Get command which is used to Get backlash compensation.

- **BA_Set**

  **Syntax**
  ```c
  int BA_Set(int axisNumber, double backlashCompensation, out string errstring)
  ```
  axisNumber: axisNumber[InCommentDoc]
  errString: The failure reason
  return: 0 in success and -1 on failure

  **Description**
  This function is used to process synchronous BA Set command which is used to Set backlash compensation.

- **BG_Get**

  **Syntax**
  ```c
  int BG_Get(int bitNumber, out string program, out string errstring)
  ```
  bitNumber: bitNumber
  program: program
  errString: The failure reason
  return: 0 in success and -1 on failure

  **Description**
  This function is used to process synchronous BG Get command which is used to Get DIO bits to execute stored programs.
ESP301 Integrated 3-Axis Motion Controller/Driver

- **BG_Set**
  
  **Syntax**
  ```c
  int BG_Set(int bitNumber, string program, out string errstring)
  bitNumber: bitNumber[InCommentDoc]
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```
  
  **Description**
  This function is used to process synchronous BG Set command which is used to Set DIO bits to execute stored programs.

- **BK_Get**
  
  **Syntax**
  ```c
  int BK_Get(int axisNumber, out int bitNumber, out int bitLevel, out string errstring)
  axisNumber: axisNumber
  bitNumber: bitNumber
  bitLevel: bitLevel
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```
  
  **Description**
  This function is used to process synchronous BK Get command which is used to Assign DIO bits to inhibit motion.

- **BK_Set**
  
  **Syntax**
  ```c
  int BK_Set(int axisNumber, int bitNumber, int bitLevel, out string errstring)
  axisNumber: axisNumber[InCommentDoc]
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```
  
  **Description**
  This function is used to process synchronous BK Set command which is used to Assign DIO bits to inhibit motion.

- **BL_Get**
  
  **Syntax**
  ```c
  int BL_Get(int axisNumber, out int value, out string errstring)
  axisNumber: axisNumber
  value: value
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```
  
  **Description**
  This function is used to process synchronous BL Get command which is used to Enable DIO bits to inhibit motion.
♦ **BL_Set**

**Syntax**
```c
int BL_Set(int axisNumber, int value, out string errstring)
axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous BL Set command which is used to Enable DIO bits to inhibit motion.

♦ **BM_Get**

**Syntax**
```c
int BM_Get(int axisNumber, out int bitNumber, out int bitLevel, out string errstring)
axisNumber: axisNumber
bitNumber: bitNumber
bitLevel: bitLevel
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous BM Get command which is used to Assign DIO bits to notify motion status.

♦ **BM_Set**

**Syntax**
```c
int BM_Set(int axisNumber, int bitNumber, int bitLevel, out string errstring)
axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous BM Set command which is used to Assign DIO bits to notify motion status.

♦ **BN_Get**

**Syntax**
```c
int BN_Get(int axisNumber, out int value, out string errstring)
axisNumber: axisNumber
value: value
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous BN Get command which is used to Enable DIO bits to notify motion status.
♦ **BN_Set**

**Syntax**
```c
int BN_Set(int axisNumber, int value, out string errstring)
axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous BN Set command which is used to Enable DIO bits to notify motion status.

♦ **BO_Get**

**Syntax**
```c
int BO_Get(out int hardwareLimitConfiguration, out string errstring)
hardwareLimitConfiguration: hardwareLimitConfiguration
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous BO Get command which is used to Set DIO port A, B, C direction.

♦ **BO_Set**

**Syntax**
```c
int BO_Set(int hardwareLimitConfiguration, out string errstring)
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous BO Set command which is used to Set DIO port A, B, C direction.

♦ **BP_Get**

**Syntax**
```c
int BP_Get(int axisNumber, out int bitNumberNeg, out int bitNumberPos, out string errstring)
axisNumber: axisNumber
bitNumberNeg: bitNumberNeg
bitNumberPos: bitNumberPos
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous BP Get command which is used to Assign DIO bits for jog mode.
♦ **BP_Set**

**Syntax**
```c
int BP_Set(int axisNumber, int bitNumberNeg, int bitNumberPos, out string errstring)
```
- `axisNumber`: axisNumber[InCommentDoc]
- `errString`: The failure reason
- `return`: 0 in success and -1 on failure

**Description**
This function is used to process synchronous BP Set command which is used to Assign DIO bits for jog mode.

♦ **BQ_Get**

**Syntax**
```c
int BQ_Get(int axisNumber, out int value, out string errstring)
```
- `axisNumber`: axisNumber
- `value`: value
- `errString`: The failure reason
- `return`: 0 in success and -1 on failure

**Description**
This function is used to process synchronous BQ Get command which is used to Enable DIO bits for jog mode.

♦ **BQ_Set**

**Syntax**
```c
int BQ_Set(int axisNumber, int value, out string errstring)
```
- `axisNumber`: axisNumber[InCommentDoc]
- `errString`: The failure reason
- `return`: 0 in success and -1 on failure

**Description**
This function is used to process synchronous BQ Set command which is used to Enable DIO bits for jog mode.

♦ **CL_Get**

**Syntax**
```c
int CL_Get(int axisNumber, out int interval, out string errstring)
```
- `axisNumber`: axisNumber
- `interval`: interval
- `errString`: The failure reason
- `return`: 0 in success and -1 on failure

**Description**
This function is used to process synchronous CL Get command which is used to Set closed loop update interval.
♦ CL_Set

**Syntax**
int CL_Set(int axisNumber, int interval, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous CL Set command which is used to set closed loop update interval.

♦ CO_Get

**Syntax**
int CO_Get(int axisNumber, out double linearCompensation, out string errstring)
axisNumber: axisNumber
linearCompensation: linearCompensation
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous CO Get command which is used to set linear compensation.

♦ CO_Set

**Syntax**
int CO_Set(int axisNumber, double linearCompensation, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous CO Set command which is used to set linear compensation.

♦ DB_Get

**Syntax**
int DB_Get(int axisNumber, out int deadBand, out string errstring)
axisNumber: axisNumber
deadBand: deadBand
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous DB Get command which is used to set position deadband.
♦ **DB_Set**

**Syntax**

```
int DB_Set(int axisNumber, int deadBand, out string errstring)
```

axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous DB Set command which is used to Set position deadband.

♦ **DC**

**Syntax**

```
int DC(int dataAcquisitionMode, int axis, int data3, int data4, int dataRate, int dataNumber, out string errstring)
```

errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous DC Set command which is used to Setup data acquisition.

♦ **DD**

**Syntax**

```
int DD(out int status, out string errstring)
```

status: status
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous DD Get command which is used to Get data acquisition done status.

♦ **DE**

**Syntax**

```
int DE(int value, out string errstring)
```

errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous DE Set command which is used to Enable/disable data acquisition.
DSP301 Integrated 3-Axis Motion Controller/Driver

- **DF**

**Syntax**
```c
int DF(out int sampleNumber, out string errstring)
```

- **sampleNumber**: sampleNumber
- **errString**: The failure reason
- **return**: 0 in success and -1 on failure

**Description**
This function is used to process synchronous DF Get command which is used to Get data acquisition sample count.

- **DG**

**Syntax**
```c
int DG(out string data, out string errstring)
```

- **data**: data
- **errString**: The failure reason
- **return**: 0 in success and -1 on failure

**Description**
This function is used to process synchronous DG Get command which is used to Get acquisition data.

- **DH_Get**

**Syntax**
```c
int DH_Get(int axisNumber, out double position, out string errstring)
```

- **axisNumber**: axisNumber
- **position**: position
- **errString**: The failure reason
- **return**: 0 in success and -1 on failure

**Description**
This function is used to process synchronous DH Get command which is used to Define home.

- **DH_Set**

**Syntax**
```c
int DH_Set(int axisNumber, double position, out string errstring)
```

- **axisNumber**: axisNumber
- **position**: position
- **errString**: The failure reason
- **return**: 0 in success and -1 on failure

**Description**
This function is used to process synchronous DH Set command which is used to Define home.
ESP301 Integrated 3-Axis Motion Controller/Driver

- **DL**

  **Syntax**
  int DL(int label, out string errstring)
  label: label [InCommentDoc]
  errString: The failure reason
  return: 0 in success and -1 on failure

  **Description**
  This function is used to process synchronous DL Set command which is used to Define label.

- **DO_Get**

  **Syntax**
  int DO_Get(int channelNumber, out double offset, out string errstring)
  channelNumber: channelNumber
  offset: offset
  errString: The failure reason
  return: 0 in success and -1 on failure

  **Description**
  This function is used to process synchronous DO Get command which is used to Set DAC offset.

- **DO_Set**

  **Syntax**
  int DO_Set(int channelNumber, double offset, out string errstring)
  channelNumber: channelNumber [InCommentDoc]
  errString: The failure reason
  return: 0 in success and -1 on failure

  **Description**
  This function is used to process synchronous DO Set command which is used to Set DAC offset.

- **DP**

  **Syntax**
  int DP(int axisNumber, out double position, out string errstring)
  axisNumber: axisNumber
  position: position
  errString: The failure reason
  return: 0 in success and -1 on failure

  **Description**
  This function is used to process synchronous DP Get command which is used to Read desired position.
♦ **DV**

**Syntax**

```c
int DV(int axisNumber, out double velocity, out string errstring)
axisNumber: axisNumber
velocity: velocity
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous DV Get command which is used to Read desired velocity.

♦ **EO_Get**

**Syntax**

```c
int EO_Get(int program, out int programNumber, out int number, out string errstring)
program: program
programNumber: programNumber
number: number
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous EO Get command which is used to Automatic execution on power on.

♦ **EO_Set**

**Syntax**

```c
int EO_Set(int program, int number, out string errstring)
program: program
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous EO Set command which is used to Automatic execution on power on.

♦ **EP**

**Syntax**

```c
int EP(int program, out string errstring)
program: program
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous EP Set command which is used to Enter program mode.
♦ **EX**

**Syntax**
```c
int EX(int program, int number, out string errstring)
```
- `program`: program
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous EX Set command which is used to execute a program.

♦ **FE_Get**

**Syntax**
```c
int FE_Get(int axisNumber, out double maxAllowedFollowingError, out string errstring)
```
- `axisNumber`: axisNumber
- `maxAllowedFollowingError`: maxAllowedFollowingError
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous FE Get command which is used to set maximum following error threshold.

♦ **FE_Set**

**Syntax**
```c
int FE_Set(int axisNumber, double maxAllowedFollowingError, out string errstring)
```
- `axisNumber`: axisNumber
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous FE Set command which is used to set maximum following error threshold.

♦ **FP_Get**

**Syntax**
```c
int FP_Get(int axisNumber, out int displayResolution, out string errstring)
```
- `axisNumber`: axisNumber
- `displayResolution`: displayResolution
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous FP Get command which is used to set position display resolution.
- **FP_Set**

  **Syntax**
  ```c
  int FP_Set(int axisNumber, int displayResolution, out string errstring)
  ```

  **Description**
  This function is used to process synchronous FP Set command which is used to Set position display resolution.

- **FR_Get**

  **Syntax**
  ```c
  int FR_Get(int axisNumber, out double encoderFullStepResolution, out string errstring)
  ```

  **Description**
  This function is used to process synchronous FR Get command which is used to Set full step resolution.

- **FR_Set**

  **Syntax**
  ```c
  int FR_Set(int axisNumber, double encoderFullStepResolution, out string errstring)
  ```

  **Description**
  This function is used to process synchronous FR Set command which is used to Set full step resolution.

- **GR_Get**

  **Syntax**
  ```c
  int GR_Get(int axisNumber, out double reductionRatio, out string errstring)
  ```

  **Description**
  This function is used to process synchronous GR Get command which is used to Set master-slave reduction ratio.
♦ GR_Set

Syntax
int GR_Set(int axisNumber, double reductionRatio, out string errstring)
axisNumber: axisNumber [InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous GR Set command which is used to Set
master-slave reduction ratio.

♦ HA_Get

Syntax
int HA_Get(int groupNumber, out double acceleration, out string errstring)
groupNumber: groupNumber
acceleration: acceleration
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous HA Get command which is used to Set
group acceleration.

♦ HA_Set

Syntax
int HA_Set(int groupNumber, double acceleration, out string errstring)
groupNumber: groupNumber [InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous HA Set command which is used to Set
group acceleration.

♦ HB

Syntax
int HB(out List<string> groups, out string errstring)
groups: groups
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous HB Get command which is used to
Read list of groups assigned.
♦ **HC_Get**

**Syntax**

```c
int HC_Get(int groupNumber, out double firstCoordinate, out double secondCoordinate, out double arcSweepAngle, out string errString)
```

groupNumber: groupNumber

firstCoordinate: firstCoordinate

secondCoordinate: secondCoordinate

arcSweepAngle: arcSweepAngle

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous HC Get command which is used to Move group along an arc.

♦ **HC_Set**

**Syntax**

```c
int HC_Set(int groupNumber, double firstCoordinate, double secondCoordinate, double arcSweepAngle, out string errString)
```

groupNumber: groupNumber

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous HC Set command which is used to Move group along an arc.

♦ **HD_Get**

**Syntax**

```c
int HD_Get(int groupNumber, out double deceleration, out string errString)
```

groupNumber: groupNumber

deceleration: deceleration

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous HD Get command which is used to Set group deceleration.

♦ **HD_Set**

**Syntax**

```c
int HD_Set(int groupNumber, double deceleration, out string errString)
```

groupNumber: groupNumber

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous HD Set command which is used to Set group deceleration.
- **HE_Get**

  **Syntax**
  ```c
  int HE_Get(int groupNumber, out double deceleration, out string errstring)
  groupNumber: groupNumber
deceleration: deceleration
erString: The failure reason
return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous HE Get command which is used to Set group e-stop deceleration.

- **HE_Set**

  **Syntax**
  ```c
  int HE_Set(int groupNumber, double deceleration, out string errstring)
  groupNumber: groupNumber
errString: The failure reason
return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous HE Set command which is used to Set group e-stop deceleration.

- **HF_Get**

  **Syntax**
  ```c
  int HF_Get(int groupNumber, out int status, out string errstring)
  groupNumber: groupNumber
status: status
erString: The failure reason
return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous HF Get command which is used to Group motor power off.

- **HF_Set**

  **Syntax**
  ```c
  int HF_Set(int groupNumber, out string errstring)
  groupNumber: groupNumber
errString: The failure reason
return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous HF Set command which is used to Group motor power off.
- **HJ_Get**

  **Syntax**
  
  int HJ_Get(int groupNumber, out double vectorJerk, out string errstring)
  
  **Description**
  
  This function is used to process synchronous HJ Get command which is used to Set group jerk.

- **HJ_Set**

  **Syntax**
  
  int HJ_Set(int groupNumber, double vectorJerk, out string errstring)
  
  **Description**
  
  This function is used to process synchronous HJ Set command which is used to Set group jerk.

- **HL_Get**

  **Syntax**
  
  int HL_Get(int groupNumber, out List<double> targets, out string errstring)
  
  **Description**
  
  This function is used to process synchronous HL Get command which is used to Move group along a line.

- **HL_Set**

  **Syntax**
  
  int HL_Set(int groupNumber, List<double> targets, out string errstring)
  
  **Description**
  
  This function is used to process synchronous HL Set command which is used to Move group along a line.
♦ HN_Get

**Syntax**
```c
int HN_Get(int groupNumber, out List<int> axes, out string errstring)
groupNumber: groupNumber
axes: axes
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous HN Get command which is used to create a new group.

♦ HN_Set

**Syntax**
```c
int HN_Set(int groupNumber, List<int> axes, out string errstring)
groupNumber: groupNumber
axes: axes
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous HN Set command which is used to create a new group.

♦ HO_Get

**Syntax**
```c
int HO_Get(int groupNumber, out int status, out string errstring)
groupNumber: groupNumber
status: status
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous HO Get command which is used to group on.

♦ HO_Set

**Syntax**
```c
int HO_Set(int groupNumber, out string errstring)
groupNumber: groupNumber
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous HO Set command which is used to group on.
ESP301 Integrated 3-Axis Motion Controller/Driver

♦ **HP**

**Syntax**
```c
int HP(int groupNumber, out List<double> positions, out string errstring)
groupNumber: groupNumber
positions: positions
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous HP Get command which is used to Read group position.

♦ **HQ_Get**

**Syntax**
```c
int HQ_Get(int groupNumber, out double bufferLevel, out string errstring)
groupNumber: groupNumber
bufferLevel: bufferLevel
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous HQ Get command which is used to Wait for group command buffer level.

♦ **HQ_Set**

**Syntax**
```c
int HQ_Set(int groupNumber, double bufferLevel, out string errstring)
groupNumber: groupNumber
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous HQ Set command which is used to Wait for group command buffer level.

♦ **HS_Get**

**Syntax**
```c
int HS_Get(int groupNumber, out int status, out string errstring)
groupNumber: groupNumber
status: status
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous HS Get command which is used to Stop group motion.
♦ **HS_Set**

**Syntax**

```c
int HS_Set(int groupNumber, out string errstring)
groupNumber: groupNumber
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**

This function is used to process synchronous HS Set command which is used to stop group motion.

♦ **HV_Get**

**Syntax**

```c
int HV_Get(int groupNumber, out double velocity, out string errstring)
groupNumber: groupNumber
velocity: velocity
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**

This function is used to process synchronous HV Get command which is used to set group velocity.

♦ **HV_Set**

**Syntax**

```c
int HV_Set(int groupNumber, double velocity, out string errstring)
groupNumber: groupNumber
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**

This function is used to process synchronous HV Set command which is used to set group velocity.

♦ **HW**

**Syntax**

```c
int HW(int groupNumber, double delay, out string errstring)
groupNumber: groupNumber
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**

This function is used to process synchronous HW Set command which is used to wait for group motion stop.
♦ HX

**Syntax**
int HX(int groupNumber, out string errstring)
groupNumber: groupNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous HX Set command which is used to Delete group.

♦ HZ

**Syntax**
int HZ(int groupNumber, out int nbAxis, out string errstring)
groupNumber: groupNumber
nbAxis: nbAxis
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous HZ Get command which is used to Read group size.

♦ ID

**Syntax**
int ID(int axisNumber, out string modelNumber, out string serialNumber, out string errstring)
axisNumber: axisNumber
modelNumber: modelNumber
serialNumber: serialNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous ID Get command which is used to Read stage model and serial number.

♦ JH_Get

**Syntax**
int JH_Get(int axisNumber, out double highSpeed, out string errstring)
axisNumber: axisNumber
highSpeed: highSpeed
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous JH Get command which is used to Set jog high speed.
♦ JH_Set

Syntax
int JH_Set(int axisNumber, double highSpeed, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous JH Set command which is used to set jog high speed.

♦ JK_Get

Syntax
int JK_Get(int axisNumber, out double jerk, out string errstring)
axisNumber: axisNumber
jerk: jerk
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous JK Get command which is used to set jerk rate.

♦ JK_Set

Syntax
int JK_Set(int axisNumber, double jerk, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous JK Set command which is used to set jerk rate.

♦ JL

Syntax
int JL(int label, int loopCount, out string errstring)
label: label
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous JL Set command which is used to Jump to label.
ESP301 Integrated 3-Axis Motion Controller/Driver

- **JW_Get**

  **Syntax**
  
  `int JW_Get(int axisNumber, out double lowSpeed, out string errstring)`
  
  `axisNumber: axisNumber`
  
  `lowSpeed: lowSpeed`
  
  `errString: The failure reason`
  
  `return: 0 in success and -1 on failure`

  **Description**
  
  This function is used to process synchronous JW Get command which is used to Set jog low speed.

- **JW_Set**

  **Syntax**
  
  `int JW_Set(int axisNumber, double lowSpeed, out string errstring)`
  
  `axisNumber: axisNumber`
  
  `errString: The failure reason`
  
  `return: 0 in success and -1 on failure`

  **Description**
  
  This function is used to process synchronous JW Set command which is used to Set jog low speed.

- **KD_Get**

  **Syntax**
  
  `int KD_Get(int axisNumber, out double derivativeGain, out string errstring)`
  
  `axisNumber: axisNumber`
  
  `derivativeGain: derivativeGain`
  
  `errString: The failure reason`
  
  `return: 0 in success and -1 on failure`

  **Description**
  
  This function is used to process synchronous KD Get command which is used to Set derivative gain.

- **KD_Set**

  **Syntax**
  
  `int KD_Set(int axisNumber, double derivativeGain, out string errstring)`
  
  `axisNumber: axisNumber`
  
  `errString: The failure reason`
  
  `return: 0 in success and -1 on failure`

  **Description**
  
  This function is used to process synchronous KD Set command which is used to Set derivative gain.
KI_Get

Syntax
int KI_Get(int axisNumber, out double integralGain, out string errstring)
axisNumber: axisNumber
integralGain: integralGain
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous KI Get command which is used to Set integral gain.

KI_Set

Syntax
int KI_Set(int axisNumber, double integralGain, out string errstring)
axisNumber: axisNumber
integralGain: integralGain
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous KI Set command which is used to Set integral gain.

KP_Get

Syntax
int KP_Get(int axisNumber, out double proportionalGain, out string errstring)
axisNumber: axisNumber
proportionalGain: proportionalGain
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous KP Get command which is used to Set proportional gain.

KP_Set

Syntax
int KP_Set(int axisNumber, double proportionalGain, out string errstring)
axisNumber: axisNumber
proportionalGain: proportionalGain
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous KP Set command which is used to Set proportional gain.
♦ **KS_Get**

**Syntax**

```c
int KS_Get(int axisNumber, out double saturationLevel, out string errstring)
```

axisNumber: axisNumber

saturationLevel: saturationLevel

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous KS Get command which is used to Set saturation level of integral factor.

♦ **KS_Set**

**Syntax**

```c
int KS_Set(int axisNumber, double saturationLevel, out string errstring)
```

axisNumber: axisNumber

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous KS Set command which is used to Set saturation level of integral factor.

♦ **LC_Get**

**Syntax**

```c
int LC_Get(out int lockOption, out string errstring)
```

lockOption: lockOption

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous LC Get command which is used to Lock/unlock keyboard.

♦ **LC_Set**

**Syntax**

```c
int LC_Set(int lockOption, out string errstring)
```

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous LC Set command which is used to Lock/unlock keyboard.
♦ **LP**

**Syntax**
```
int LP(int program, out List<string> programs, out string errstring)
```
- `program`: program
- `programs`: programs
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous LP Get command which is used to List program.

♦ **MD**

**Syntax**
```
int MD(int axisNumber, out int status, out string errstring)
```
- `axisNumber`: axisNumber
- `status`: status
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous MD Get command which is used to Read motion done status.

♦ **MF_Get**

**Syntax**
```
int MF_Get(int axisNumber, out int status, out string errstring)
```
- `axisNumber`: axisNumber
- `status`: status
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous MF Get command which is used to Motor power off.

♦ **MF_Set**

**Syntax**
```
int MF_Set(int axisNumber, out string errstring)
```
- `axisNumber`: axisNumber
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous MF Set command which is used to Motor power off.
♦ **MO_Get**

**Syntax**

```c
int MO_Get(int axisNumber, out int status, out string errstring)
```

- `axisNumber`: axisNumber
- `status`: status
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous MO Get command which is used to Motor power on.

♦ **MO_Set**

**Syntax**

```c
int MO_Set(int axisNumber, out string errstring)
```

- `axisNumber`: axisNumber
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous MO Set command which is used to Motor power on.

♦ **MT_Get**

**Syntax**

```c
int MT_Get(int axisNumber, out int status, out string errstring)
```

- `axisNumber`: axisNumber
- `status`: status
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous MT Get command which is used to Move to hardware travel limit.

♦ **MT_Set**

**Syntax**

```c
int MT_Set(int axisNumber, string direction, out string errstring)
```

- `axisNumber`: axisNumber
- `errString`: The failure reason
- return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous MT Set command which is used to Move to hardware travel limit.
♦ MV_Get

Syntax
int MV_Get(int axisNumber, out int status, out string errString)
axisNumber: axisNumber
status: status
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous MV Get command which is used to Move indefinitely.

♦ MV_Set

Syntax
int MV_Set(int axisNumber, string direction, out string errString)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous MV Set command which is used to Move indefinitely.

♦ MZ_Get

Syntax
int MZ_Get(int axisNumber, out int status, out string errString)
axisNumber: axisNumber
status: status
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous MZ Get command which is used to Move to nearest index.

♦ MZ_Set

Syntax
int MZ_Set(int axisNumber, string direction, out string errString)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous MZ Set command which is used to Move to nearest index.
♦ **OH_Get**

**Syntax**
```c
int OH_Get(int axisNumber, out double highSpeed, out string errstring)
axisNumber: axisNumber
highSpeed: highSpeed
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous OH Get command which is used to Set home search high speed.

♦ **OH_Set**

**Syntax**
```c
int OH_Set(int axisNumber, double highSpeed, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous OH Set command which is used to Set home search high speed.

♦ **OL_Get**

**Syntax**
```c
int OL_Get(int axisNumber, out double lowSpeed, out string errstring)
axisNumber: axisNumber
lowSpeed: lowSpeed
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous OL Get command which is used to Set home search low speed.

♦ **OL_Set**

**Syntax**
```c
int OL_Set(int axisNumber, double lowSpeed, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous OL Set command which is used to Set home search low speed.
♦ **OM_Get**

**Syntax**

```c
int OM_Get(int axisNumber, out int mode, out string errstring)
```

axisNumber: axisNumber
mode: mode
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous OM Get command which is used to Set home search mode.

♦ **OM_Set**

**Syntax**

```c
int OM_Set(int axisNumber, int mode, out string errstring)
```

axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous OM Set command which is used to Set home search mode.

♦ **OR**

**Syntax**

```c
int OR(int axisNumber, int mode, out string errstring)
```

axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous OR Set command which is used to Search for home.

♦ **PA_Get**

**Syntax**

```c
int PA_Get(int axisNumber, out double position, out string errstring)
```

axisNumber: axisNumber
position: position
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous PA Get command which is used to Move to absolute position.
♦ **PA_Set**

**Syntax**
```
int PA_Set(int axisNumber, double position, out string errstring)
```

- `axisNumber`: axisNumber [InCommentDoc]
- `errString`: The failure reason
- Return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous PA Set command which is used to Move to absolute position.

♦ **PH**

**Syntax**
```
int PH(out int status1, out int status2, out string errstring)
```

- `status1`: status1
- `status2`: status2
- `errString`: The failure reason
- Return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous PH Get command which is used to Get hardware status.

♦ **PR**

**Syntax**
```
int PR(int axisNumber, double increment, out string errstring)
```

- `axisNumber`: axisNumber [InCommentDoc]
- `errString`: The failure reason
- Return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous PR Set command which is used to Move to relative position.

♦ **QD**

**Syntax**
```
int QD(int axisNumber, out string errstring)
```

- `axisNumber`: axisNumber [InCommentDoc]
- `errString`: The failure reason
- Return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QD Set command which is used to Update motor driver settings.
♦ QG_Get

**Syntax**
int QG_Get(int axisNumber, out double gearConstant, out string errstring)
axisNumber: axisNumber
gearConstant: gearConstant
erString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QG Get command which is used to Set gear constant.

♦ QG_Set

**Syntax**
int QG_Set(int axisNumber, double gearConstant, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QG Set command which is used to Set gear constant.

♦ QI_Get

**Syntax**
int QI_Get(int axisNumber, out double motorCurrent, out string errstring)
axisNumber: axisNumber
motorCurrent: motorCurrent
erString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QI Get command which is used to Set maximum motor current.

♦ QI_Set

**Syntax**
int QI_Set(int axisNumber, double motorCurrent, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QI Set command which is used to Set maximum motor current.
*QM_Get*

**Syntax**

```c
int QM_Get(int axisNumber, out int motorType, out string errstring)
```

- `axisNumber`: axisNumber
- `motorType`: motorType
- `errString`: The failure reason
- `return`: 0 in success and -1 on failure

**Description**

This function is used to process synchronous QM Get command which is used to Set motor type.

*QM_Set*

**Syntax**

```c
int QM_Set(int axisNumber, int motorType, out string errstring)
```

- `axisNumber`: axisNumber
- `motorType`: motorType
- `errString`: The failure reason
- `return`: 0 in success and -1 on failure

**Description**

This function is used to process synchronous QM Set command which is used to Set motor type.

*QP*

**Syntax**

```c
int QP(out string errstring)
```

- `errString`: The failure reason
- `return`: 0 in success and -1 on failure

**Description**

This function is used to process synchronous QP Set command which is used to Quit program mode.

*QR_Get*

**Syntax**

```c
int QR_Get(int axisNumber, out int delay, out int reductionPercentage, out string errstring)
```

- `axisNumber`: axisNumber
- `delay`: delay
- `reductionPercentage`: reductionPercentage
- `errString`: The failure reason
- `return`: 0 in success and -1 on failure

**Description**

This function is used to process synchronous QR Get command which is used to Reduce motor torque.
♦ QR_Set

**Syntax**
int QR_Set(int axisNumber, int delay, int reductionPercentage, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QR Set command which is used to Reduce motor torque.

♦ QS_Get

**Syntax**
int QS_Get(int axisNumber, out int microStep, out string errstring)
axisNumber: axisNumber
microStep: microStep
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QS Get command which is used to Set microstep factor.

♦ QS_Set

**Syntax**
int QS_Set(int axisNumber, int microStep, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QS Set command which is used to Set microstep factor.

♦ QT_Get

**Syntax**
int QT_Get(int axisNumber, out double tachometerGain, out string errstring)
axisNumber: axisNumber
tachometerGain: tachometerGain
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous QT Get command which is used to Set tachometer gain.
 nhãn QT_Set
Syntax
int QT_Set(int axisNumber, double tachometerGain, out string errstring)
axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous QT Set command which is used to Set tachometer gain.

 nhãn QV_Get
Syntax
int QV_Get(int axisNumber, out double motorVoltage, out string errstring)
axisNumber: axisNumber
motorVoltage: motorVoltage
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous QV Get command which is used to Set average motor voltage.

 nhãn QV_Seq
Syntax
int QV_Seq(int axisNumber, double motorVoltage, out string errstring)
axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous QV Set command which is used to Set average motor voltage.

 nhãn RQ
Syntax
int RQ(int interruptNumber, out string errstring)
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous RQ Set command which is used to Generate service request.
♦ **RS**

**Syntax**

```
int RS(out string errstring)
```

errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous RS Set command which is used to Reset the controller.

♦ **SA_Get**

**Syntax**

```
int SA_Get(out int adress, out string errstring)
```

adress: adress
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous SA Get command which is used to Set device address.

♦ **SA_Set**

**Syntax**

```
int SA_Set(int adress, out string errstring)
```

errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous SA Set command which is used to Set device address.

♦ **SB_Get**

**Syntax**

```
int SB_Get(out int hardwareLimitConfiguration, out string errstring)
```

hardwareLimitConfiguration: hardwareLimitConfiguration
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous SB Get command which is used to Set/get DIO port A, B, C bit status.
♦ SB_Set

**Syntax**
```
int SB_Set(int hardwareLimitConfiguration, out string errstring)
```
**errString:** The failure reason
**return:** 0 in success and -1 on failure

**Description**
This function is used to process synchronous SB Set command which is used to set/get DIO port A, B, C bit status.

♦ SH_Get

**Syntax**
```
int SH_Get(int axisNumber, out double homePresetPosition, out string errstring)
```
**axisNumber:** axisNumber
**homePresetPosition:** homePresetPosition
**errString:** The failure reason
**return:** 0 in success and -1 on failure

**Description**
This function is used to process synchronous SH Get command which is used to set home preset position.

♦ SH_Set

**Syntax**
```
int SH_Set(int axisNumber, double homePresetPosition, out string errstring)
```
**axisNumber:** axisNumber
**homePresetPosition:** homePresetPosition
**errString:** The failure reason
**return:** 0 in success and -1 on failure

**Description**
This function is used to process synchronous SH Set command which is used to set home preset position.

♦ SI_Get

**Syntax**
```
int SI_Get(out int velocity, out string errstring)
```
**velocity:** velocity
**errString:** The failure reason
**return:** 0 in success and -1 on failure

**Description**
This function is used to process synchronous SI Get command which is used to set master-slave jog velocity update interval.
♦ **SI_Set**

**Syntax**
int SI_Set(int velocity, out string errstring)
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SI Set command which is used to set master-slave jog velocity update interval.

♦ **SK_Get**

**Syntax**
int SK_Get(out double coefficient1, out double coefficient2, out string errstring)
coefficient1: coefficient1
coefficient2: coefficient2
erString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SK Get command which is used to set master-slave jog velocity scaling coefficients.

♦ **SK_Set**

**Syntax**
int SK_Set(double coefficient1, double coefficient2, out string errstring)
erString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SK Set command which is used to set master-slave jog velocity scaling coefficients.

♦ **SL_Get**

**Syntax**
int SL_Get(int axisNumber, out double limit, out string errstring)
axisNumber: axisNumber
limit: limit
erString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SL Get command which is used to set level travel limit.
♦ SL_Set

Syntax
int SL_Set(int axisNumber, double limit, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous SL Set command which is used to Set
level travel limit.

♦ SM

Syntax
int SM(out string errstring)
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous SM Set command which is used to
Save settings to non-volatile memory.

♦ SN_Get

Syntax
int SN_Get(int axisNumber, out int unit, out string errstring)
axisNumber: axisNumber
unit: unit
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous SN Get command which is used to Set
axis displacement units.

♦ SN_Set

Syntax
int SN_Set(int axisNumber, int unit, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous SN Set command which is used to Set
axis displacement units.
♦ **SR_Get**

**Syntax**
int SR_Get(int axisNumber, out double limit, out string errstring)
axisNumber: axisNumber
limit: limit
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SR Get command which is used to Set right travel limit.

♦ **SR_Set**

**Syntax**
int SR_Set(int axisNumber, double limit, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SR Set command which is used to Set right travel limit.

♦ **SS_Get**

**Syntax**
int SS_Get(int axisNumber, out int masterAxis, out string errstring)
axisNumber: axisNumber
masterAxis: masterAxis
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SS Get command which is used to Define master-slave relationship.

♦ **SS_Set**

**Syntax**
int SS_Set(int axisNumber, int masterAxis, out string errstring)
axisNumber: axisNumber
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SS Set command which is used to Define master-slave relationship.
♢ **ST**

**Syntax**
int ST(int axisNumber, out string errstring)

axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous ST Set command which is used to Stop motion.

♢ **SU_Get**

**Syntax**
int SU_Get(int axisNumber, out double encoderResolution, out string errstring)

axisNumber: axisNumber
encoderResolution: encoderResolution
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SU Get command which is used to Set encoder resolution.

♢ **SU_Set**

**Syntax**
int SU_Set(int axisNumber, double encoderResolution, out string errstring)

axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous SU Set command which is used to Set encoder resolution.

♢ **TB**

**Syntax**
int TB(out string errorCode, out string timestamp, out string errorMessage, out string errstring)

errorCode: errorCode
timestamp: timestamp
errorMessage: errorMessage
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous TB Get command which is used to Read error message.
ESP301 Integrated 3-Axis Motion Controller/Driver

- **TE**

  **Syntax**
  ```
  int TE(out string errorCode, out string errstring)
  errorCode: errorCode
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous TE Get command which is used to Read error code.

- **TJ_Get**

  **Syntax**
  ```
  int TJ_Get(int axisNumber, out int homeMode, out string errstring)
  axisNumber: axisNumber
  homeMode: homeMode
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous TJ Get command which is used to Set trajectory mode.

- **TJ_Set**

  **Syntax**
  ```
  int TJ_Set(int axisNumber, int homeMode, out string errstring)
  axisNumber: axisNumber
  homeMode: homeMode
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous TJ Set command which is used to Set trajectory mode.

- **TP**

  **Syntax**
  ```
  int TP(int axisNumber, out double position, out string errstring)
  axisNumber: axisNumber
  position: position
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous TP Get command which is used to Read actual position.
ESP301 Integrated 3-Axis Motion Controller/Driver

♦ TS

**Syntax**

int TS(out status, out string errstring)
status: status
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous TS Get command which is used to Get controller status.

♦ TV

**Syntax**

int TV(int axisNumber, out double velocity, out string errstring)
axisNumber: axisNumber
velocity: velocity
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous TV Get command which is used to Get actual velocity.

♦ TX

**Syntax**

int TX(out status, out string errstring)
status: status
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous TX Get command which is used to Get controller activity.

♦ UF

**Syntax**

int UF(out string errstring)
errString: The failure reason
return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous UF Set command which is used to Update servo filter.
♦ UF

**Syntax**
```c
int UF(int axisNumber, out string errstring)
axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous UF Set command which is used to Update servo filter.

♦ UH

**Syntax**
```c
int UH(int bitNumber, out string errstring)
bitNumber: bitNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous UH Set command which is used to Wait for DIO bit high.

♦ UL

**Syntax**
```c
int UL(int bitNumber, out string errstring)
bitNumber: bitNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous UL Set command which is used to Wait for DIO bit low.

♦ VA_Get

**Syntax**
```c
int VA_Get(int axisNumber, out double velocity, out string errstring)
axisNumber: axisNumber
velocity: velocity
errString: The failure reason
return: 0 in success and -1 on failure
```

**Description**
This function is used to process synchronous VA Get command which is used to Set velocity.
VA_Set

Syntax
int VA_Set(int axisNumber, double velocity, out string errstring)
axisNumber: axisNumber[InCommentDoc]
erString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous VA Set command which is used to Set velocity.

VB_Get

Syntax
int VB_Get(int axisNumber, out double baseVelocity, out string errstring)
axisNumber: axisNumber
baseVelocity: baseVelocity
erString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous VB Get command which is used to Set base velocity for step motors.

VB_Set

Syntax
int VB_Set(int axisNumber, double baseVelocity, out string errstring)
axisNumber: axisNumber[InCommentDoc]
erString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous VB Set command which is used to Set base velocity for step motors.

VE

Syntax
int VE(out string controllerVersion, out string errstring)
controllerVersion: controllerVersion
erString: The failure reason
return: 0 in success and -1 on failure

Description
This function is used to process synchronous VE Get command which is used to Read controller firmware version.
♦ **VF_Get**

**Syntax**
```
int VF_Get(int axisNumber, out double velocityFeedForwardGain, out string errstring)
```
axisNumber: axisNumber
velocityFeedForwardGain: velocityFeedForwardGain
erString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous VF Get command which is used to Set velocity feed-forward gain.

♦ **VF_Set**

**Syntax**
```
int VF_Set(int axisNumber, double velocityFeedForwardGain, out string errstring)
```
axisNumber: axisNumber
velocityFeedForwardGain: velocityFeedForwardGain
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous VF Set command which is used to Set velocity feed-forward gain.

♦ **VU_Get**

**Syntax**
```
int VU_Get(int axisNumber, out double maxVelocity, out string errstring)
```
axisNumber: axisNumber
maxVelocity: maxVelocity
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous VU Get command which is used to Set maximum velocity.

♦ **VU_Set**

**Syntax**
```
int VU_Set(int axisNumber, double maxVelocity, out string errstring)
```
axisNumber: axisNumber
maxVelocity: maxVelocity
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous VU Set command which is used to Set maximum velocity.
ESP301 Integrated 3-Axis Motion Controller/Driver

- **WP**

  **Syntax**
  
  int WP(int axisNumber, double position, out string errstring)
  
  axisNumber: axisNumber[InCommentDoc]
  
  errString: The failure reason
  
  return: 0 in success and -1 on failure

  **Description**
  
  This function is used to process synchronous WP Set command which is used to Wait for absolute position crossing.

- **WS**

  **Syntax**
  
  int WS(int axisNumber, int delay, out string errstring)
  
  axisNumber: axisNumber[InCommentDoc]
  
  errString: The failure reason
  
  return: 0 in success and -1 on failure

  **Description**
  
  This function is used to process synchronous WS Set command which is used to Wait for motion stop.

- **WT**

  **Syntax**
  
  int WT(int delay, out string errstring)
  
  errString: The failure reason
  
  return: 0 in success and -1 on failure

  **Description**
  
  This function is used to process synchronous WT Set command which is used to Wait.

- **XM**

  **Syntax**
  
  int XM(out string availableStorageSpace, out string errstring)
  
  availableStorageSpace: availableStorageSpace
  
  errString: The failure reason
  
  return: 0 in success and -1 on failure

  **Description**
  
  This function is used to process synchronous XM Get command which is used to Get available program memory.
ESP301 Integrated 3-Axis Motion Controller/Driver

❖ XX

**Syntax**
int XX(int program, out string errstring)
program: program[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous XX Set command which is used to Delete a stored program.

❖ ZA_Get

**Syntax**
int ZA_Get(int axisNumber, out int amplifierIIOConfiguration, out string errstring)
axisNumber: axisNumber
amplifierIIOConfiguration: amplifierIIOConfiguration
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous ZA Get command which is used to Set amplifier I/O configuration.

❖ ZA_Set

**Syntax**
int ZA_Set(int axisNumber, int amplifierIIOConfiguration, out string errstring)
axisNumber: axisNumber[InCommentDoc]
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous ZA Set command which is used to Set amplifier I/O configuration.

❖ ZB_Get

**Syntax**
int ZB_Get(int axisNumber, out int feedbackConfiguration, out string errstring)
axisNumber: axisNumber
feedbackConfiguration: feedbackConfiguration
errString: The failure reason
return: 0 in success and -1 on failure

**Description**
This function is used to process synchronous ZB Get command which is used to Set feedback configuration.
- **ZB_Set**

  **Syntax**
  ```
  int ZB_Set(int axisNumber, int feedbackConfiguration, out string errstring)
  axisNumber: axisNumber[InCommentDoc]
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous ZB Set command which is used to Set feedback configuration.

- **ZE_Get**

  **Syntax**
  ```
  int ZE_Get(int axisNumber, out int estopConfiguration, out string errstring)
  axisNumber: axisNumber
  estopConfiguration: estopConfiguration
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous ZE Get command which is used to Set E-stop configuration.

- **ZE_Set**

  **Syntax**
  ```
  int ZE_Set(int axisNumber, int estopConfiguration, out string errstring)
  axisNumber: axisNumber[InCommentDoc]
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous ZE Set command which is used to Set E-stop configuration.

- **ZF_Get**

  **Syntax**
  ```
  int ZF_Get(int axisNumber, out int followingErrorConfiguration, out string errstring)
  axisNumber: axisNumber
  followingErrorConfiguration: followingErrorConfiguration
  errString: The failure reason
  return: 0 in success and -1 on failure
  ```

  **Description**
  This function is used to process synchronous ZF Get command which is used to Set following error configuration.
♦ **ZF_Set**

**Syntax**

```c
int ZF_Set(int axisNumber, int followingErrorConfiguration, out string errstring)
```

- **axisNumber**: axisNumber
- **errString**: The failure reason
- **return**: 0 in success and -1 on failure

**Description**

This function is used to process synchronous ZF Set command which is used to Set following error configuration.

♦ **ZH_Get**

**Syntax**

```c
int ZH_Get(int axisNumber, out int hardwareLimitConfiguration, out string errstring)
```

- **axisNumber**: axisNumber
- **hardwareLimitConfiguration**: hardwareLimitConfiguration
- **errString**: The failure reason
- **return**: 0 in success and -1 on failure

**Description**

This function is used to process synchronous ZH Get command which is used to Set hardware limit configuration.

♦ **ZH_Set**

**Syntax**

```c
int ZH_Set(int axisNumber, int hardwareLimitConfiguration, out string errstring)
```

- **axisNumber**: axisNumber
- **errString**: The failure reason
- **return**: 0 in success and -1 on failure

**Description**

This function is used to process synchronous ZH Set command which is used to Set hardware limit configuration.

♦ **ZS_Get**

**Syntax**

```c
int ZS_Get(int axisNumber, out int softwareLimitConfiguration, out string errstring)
```

- **axisNumber**: axisNumber
- **softwareLimitConfiguration**: softwareLimitConfiguration
- **errString**: The failure reason
- **return**: 0 in success and -1 on failure

**Description**

This function is used to process synchronous ZS Get command which is used to Set software limit configuration.
♦ **ZS_Set**

**Syntax**

```c
int ZS_Set(int axisNumber, int softwareLimitConfiguration, out string errstring)
```

axisNumber: `axisNumber`[InCommentDoc]

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous ZS Set command which is used to Set software limit configuration.

♦ **ZU**

**Syntax**

```c
int ZU(out int espSystemConfiguration, out string errstring)
```

espSystemConfiguration: `espSystemConfiguration`

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous ZU Get command which is used to Get ESP system configuration.

♦ **ZZ_Get**

**Syntax**

```c
int ZZ_Get(out int systemConfiguration, out string errstring)
```

systemConfiguration: `systemConfiguration`

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous ZZ Get command which is used to Set system configuration.

♦ **ZZ_Set**

**Syntax**

```c
int ZZ_Set(int systemConfiguration, out string errstring)
```

errString: The failure reason

return: 0 in success and -1 on failure

**Description**

This function is used to process synchronous ZZ Set command which is used to Set system configuration.
3.0 Python Example

```python
#===================================================================== # Initialization Start # The script within Initialization Start and Initialization End # is needed for properly initializing Command Interface for ESP301 instrument. # The user should copy this code as is and specify correct paths here. import sys

# Command Interface DLL can be found here. print "Adding location of Newport.ESP301.CommandInterface.dll to sys.path" sys.path.append(r' C:\Program Files \x86\Newport\MotionControl\ESP301\Bin) # The CLR module provide functions for interacting with the underlying # .NET runtime
import clr

# Add reference to assembly and import names from namespace clr.AddReferenceToFile("Newport.ESP301.CommandInterface.dll") from CommandInterface import *

import System
#===================================================================== # Instrument Initialization # The key should have double slashes since # (one of them is escape character) instrument="COM15" BAUDRATE = 921600 print 'Instrument Key=>', instrument

# create an ESP301 instance ESP301Device = ESP301()

# Open communication ret = esp301.OpenInstrument(instrument, BAUDRATE);

# Get positive software limit result, response, errString = ESP301Device.SR_Get(1) if result == 0 : print 'positive software limit=>', response else: print 'Error=>',errString

# Get negative software limit result, response, errString = ESP301Device.SL_Get(1) if result == 0 : print 'negative software limit=>', response else: print 'Error=>',errString

# Get controller revision information result, response, errString = ESP301Device.VE() if result == 0 :```

```
print 'controller revision=>', response
else:
    print 'Error=>', errString
# Get current position
result, response, errString = esp301Device.TP(1)
if result == 0:
    print 'position=>', response
else:
    print 'Error=>', errString

# Close communication
esp301.CloseInstrument();
Service Form

Name: __________________________________________________  Return authorization #: __________________

Company: _______________________________________________  (Please obtain prior to return of item)

Address: ________________________________________________  Date: ___________________________

Country: ________________________________________________  Phone Number: ______________________

P.O. Number: ____________________________________________  Fax Number: _________________________

Item(s) Being Returned: ____________________________________  Model#: ____________________________

Description: _______________________________________________________________________________

Reasons of return of goods (please list any specific problems):

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

___________________________________________________________________________________________

_________________________________________________________________________________________
North America & Asia  
Newport Corporation  
1791 Deere Ave.  
Irvine, CA 92606, USA  
**Sales**  
Tel.: (800) 222-6440  
e-mail: sales@newport.com  
**Technical Support**  
Tel.: (800) 222-6440  
e-mail: tech@newport.com  
**Service, RMAs & Returns**  
Tel.: (800) 222-6440  
e-mail: service@newport.com

Europe  
MICRO-CONTROLE Spectra-Physics S.A.S  
9, rue du Bois Sauvage  
91055 Évry CEDEX  
France  
**Sales**  
Tel.: +33 (0)1.60.91.68.68  
e-mail: france@newport.com  
**Technical Support**  
e-mail: tech_europe@newport.com  
**Service & Returns**  
Tel.: +33 (0)2.38.40.51.55