Nexo TCP Socket JSON Interface

Author: Eckhardt Feyhl, DC-AT/ESJ12, Bosch-Rexroth AG, Murrhardt, 2. May 2019

A short introduction over Nexo's JSON interface.

Content

Connection

The cordless nutrunner tool Nexo provides a native TCP socket server on **port 64121**.

Up to 3 connections are simultaneous possible. The 4th connection will shut down the oldest connection.

There is no time out and no keep alive implemented.

Basic of this communication interface is a simple request and response protocol in JSON format.

Important: Please send the JSON stream including the terminating zero character. This speed ups the communication.

JSON tag definitions

cmdID	String	the content of this tag will be copied into the tag retOfCmdID of the response. This can be used by the sender of a JSON command to identify the response.
retOfCmdID	String	will be used in the response JSON stream. It contains the content of the calling cmdID tag.
comp	String	The functions are categorized by components. See following function description.
fct	String	The name oft he function to be called. See following function description.

Following sections explains the implemented functions in the defined components.

Tightening Results

Each tightening result is stored with a unique cycle number. Per default up to 1000 results can be stored on Nexo's data base. With the cycle number you can get all data of a tightening result incl. the curve data in JSON format.

availableCycles

Following command returns the min. and max. value of the available cycle numbers and the total number of available results:

Request:

```
{
"cmdID":"this will be copied into retOfCmdID",
"comp":"TighteningResults",
"fct":"availableCycles"
}
```

The response of this JSON command can look like this:

```
{
"retOfCmdID":"this will be copied into retOfCmdID",
"minCycle":44512,
"maxCycle":45457,
"availableResults":943
}
```

minCycle	Number	contains the cycle number of the oldest tightening result
maxCycle	Number	contains the cycle number of the youngest tightening result
availableResults	Number	contains the number of available tightening results. NOTICE: the number of available tightening results can be different to the calculated number of results (maxCycle minus minCycle + 1). Normaly the oldest result will be deleted if the total number of results exceeds the limit. But if this result is marked as „unsend“ another result will be deleted instead. This means that the cycle numbers are not strongly ascending by 1.

result

This function returns a tightening result with the given cycle number. This call can take up to 300ms.

Request:

```
{
"cmdID":"command number 123",
"comp":"TighteningResults",
"fct":"result",
"cycle":45457
}
```

If **cycle** does not exist the rslt tag will be empty. See also **minCycle** and **maxCycle** of availableCycles.

Response:

```
{
"retOfCmdID":"command number 123",
"rslt":{
        "nr":  1,
        "result":       "OK",
        "channel":      "NxChNameEF",
        "prg nr":       1,
        "prg name":     "prg01_short",
        "prg date":     "2017-06-13 15:22:09",
        "cycle":    45457,
        "nominal torque":       12,
        "date": "2000-01-01 21:21:36",
        "id code":      "PI1k1",
        "torque unit": "Nm",
        "last cmd":     "TF Angle",
        "quality code":         "1",
        "total time":   "0.072000",
        "tool serial": "311000341",
        "rework code": 0,
        "rework text": "",
        "cell id":      "0000",
        "job nr":       -1,
        "MCE factor":   1,
        "batch nr":     "1",
        "batch canceled":       0,
        "batch direction OK":   1,
        "batch direction NOK":1,
        "batch max OK":         99,
        "batch max NOK":        100,
        "batch OK":     1,
        "batch NOK":    0,
        "tightening steps":     [{
                "result":       "OK",
                "name": "short",
                "step type":    "standard",
                "row":  "2",
                "column":       "A",
                "category":     1,
                "last cmd":     "TF Angle",
                "torque":       0.030000,
                "angle":        38.560000,
                "duration":     0.072000,
                "quality code":         "1",
                "speed":        180,
                "angle threshold nom":0,
                "angle threshold act":0.005000,
                "tightening functions":         [{
                        "name": "TF Angle",
                        "nom":  30,
                        "act":  30.580000
                }, {
                        "name": "MFs TimeMax",
                        "nom":  1,
```

```
                        "act": 0.072000
                }, {
                        "name": "MF TorqueMin",
                        "nom": -1,
                        "act": 0.030000
                }, {
                        "name": "MF TorqueMax",
                        "nom": 2,
                        "act": 0.030000
                }, {
                        "name": "MF AngleMin",
                        "nom": 0,
                        "act": 38.560000
                }, {
                        "name": "MF AngleMax",
                        "nom": 60,
                        "act": 38.560000
                }],
                "graph":        {
                "angle values":         [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.660000, 0.660000,
1.330000, 1.990000, 3.320000, 3.990000, 5.320000, 6.650000, 7.310000, 8.640000, 9.970000,
11.300000, 12.630000, 13.960000, 15.290000, 15.960000, 17.290000, 18.610000, 19.940000,
21.270000, 22.600000, 23.930000, 24.600000, 25.930000, 27.260000, 28.590000, 29.920000,
30.580000, 31.910000, 33.240000, 34.570000, 35.240000, 35.900000, 36.570000, 37.230000,
37.890000, 38.560000, 38.560000],
                "torque values":        [0.005000, 0.005000, 0.005000, 0.005000, 0.005000,
0.005000, 0.005000, 0.005000, 0.005000, 0.013000, 0.005000, 0.005000, 0.013000, 0.013000,
0.013000, 0.013000, 0.013000, 0.013000, 0.013000, 0.013000, 0.013000, 0.056000, 0.115000,
0.030000, 0.030000, 0.030000, 0.039000, 0.039000, 0.022000, 0.030000, 0.039000, 0.030000,
0.022000, 0.022000, 0.022000, 0.022000, 0.013000, 0.022000, 0.022000, 0.030000, 0.030000,
0.022000, 0.022000, 0.005000, 0.005000, 0.005000, 0.013000, 0.013000, 0.010000, 0.006000],
                "time values": [0, 0.002000, 0.003000, 0.004000, 0.005000, 0.006000, 0.007000,
0.008000, 0.009000, 0.010000, 0.011000, 0.012000, 0.013000, 0.014000, 0.015000, 0.016000,
0.017000, 0.018000, 0.019000, 0.020000, 0.021000, 0.022000, 0.023000, 0.024000, 0.025000,
0.026000, 0.027000, 0.028000, 0.029000, 0.030000, 0.031000, 0.032000, 0.033000, 0.034000,
0.035000, 0.036000, 0.037000, 0.038000, 0.039000, 0.040000, 0.041000, 0.042000, 0.043000,
0.044000, 0.045000, 0.046000, 0.047000, 0.048000, 0.051000, 0.072000]
                        }
                }]
        }
}
```


`rslt` Object is a JSON object with the same content like Nexo's ftp or http protocol. For more informations please refer to chapter 9.8.4.2 in Project planning document 3608870A47_AE_EN_NEXO-V1300.pdf.

Notice: **`rslt`** can be empty if the cycle number does not exist. See also function availableCycles.

subscribeResults

to get the tightening results automatically

Request:

```
{
"cmdID":"124",
"comp":"TighteningResults",
"fct":"subscribeResults"
}
```

Response:

```
{
"retOfCmdID": "124",
"rslt": true
}
```

The automatical sent tightening result is the same as Nexo's ftp or http protocol e.g.:

```
{
"nr":  1,
"result":      "OK",
"channel":     "NxChNameEF",
"prg nr":      1,
"prg name":    "prg01_short",
"prg date":    "2017-06-13 15:22:09",
"cycle":       45457,
"nominal torque":     12,
"date":"2000-01-01 21:21:36",
"id code":     "PI1k1",
"torque unit": "Nm",
"last cmd":    "TF Angle",
"quality code":       "1",
"total time":  "0.072000",
"tool serial": "311000341",
"rework code": 0,
"rework text": "",
"cell id":     "0000",
"job nr":      -1,
"MCE factor":  1,
"batch nr":    "1",
"batch canceled":     0,
"batch direction OK": 1,
"batch direction NOK":1,
"batch max OK":       99,
"batch max NOK":      100,
"batch OK":    1,
"batch NOK":   0,
"tightening steps":   [{
        "result":      "OK",
        "name":"short",
        "step type":   "standard",
        "row":  "2",
        "column":      "A",
        "category":    1,
        "last cmd":    "TF Angle",
        "torque":      0.030000,
        "angle":       38.560000,
        "duration":    0.072000,
        "quality code":       "1",
        "speed":       180,
        "angle threshold nom":0,
        "angle threshold act":0.005000,
        "tightening functions":       [{
                "name":"TF Angle",
                "nom":  30,
                "act":  30.580000
        }, {
                "name":"MFs TimeMax",
                "nom":  1,
                "act":  0.072000
        }, {
                "name":"MF TorqueMin",
```

```
                "nom": -1,
                "act": 0.030000
        }, {
                "name":"MF TorqueMax",
                "nom": 2,
                "act": 0.030000
        }, {
                "name":"MF AngleMin",
                "nom": 0,
                "act": 38.560000
        }, {
        "name":"MF AngleMax",
                "nom": 60,
                "act": 38.560000
        }],
        "graph":        {
        "angle values":         [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.660000, 0.660000,
1.330000, 1.990000, 3.320000, 3.990000, 5.320000, 6.650000, 7.310000, 8.640000, 9.970000,
11.300000, 12.630000, 13.960000, 15.290000, 15.960000, 17.290000, 18.610000, 19.940000,
21.270000, 22.600000, 23.930000, 24.600000, 25.930000, 27.260000, 28.590000, 29.920000,
30.580000, 31.910000, 33.240000, 34.570000, 35.240000, 35.900000, 36.570000, 37.230000,
37.890000, 38.560000, 38.560000],
        "torque values":        [0.005000, 0.005000, 0.005000, 0.005000, 0.005000, 0.005000,
0.005000, 0.005000, 0.005000, 0.013000, 0.005000, 0.005000, 0.013000, 0.013000, 0.013000,
0.013000, 0.013000, 0.013000, 0.013000, 0.013000, 0.013000, 0.056000, 0.115000, 0.030000,
0.030000, 0.030000, 0.039000, 0.039000, 0.022000, 0.030000, 0.039000, 0.030000, 0.022000,
0.022000, 0.022000, 0.022000, 0.013000, 0.022000, 0.022000, 0.030000, 0.030000, 0.022000,
0.022000, 0.005000, 0.005000, 0.005000, 0.013000, 0.013000, 0.010000, 0.006000],
        "time values": [0, 0.002000, 0.003000, 0.004000, 0.005000, 0.006000, 0.007000,
0.008000, 0.009000, 0.010000, 0.011000, 0.012000, 0.013000, 0.014000, 0.015000, 0.016000,
0.017000, 0.018000, 0.019000, 0.020000, 0.021000, 0.022000, 0.023000, 0.024000, 0.025000,
0.026000, 0.027000, 0.028000, 0.029000, 0.030000, 0.031000, 0.032000, 0.033000, 0.034000,
0.035000, 0.036000, 0.037000, 0.038000, 0.039000, 0.040000, 0.041000, 0.042000, 0.043000,
0.044000, 0.045000, 0.046000, 0.047000, 0.048000, 0.051000, 0.072000]
                }
        }]
}
```


unsubscribeResults

To stop the automatical send of tightening results.

Request:

```
{
"cmdID":"125",
"comp":"TighteningResults",
"fct":"unsubscribeResults"
}
```

Response:

```
{
"retOfCmdID":"125",
"rslt":true
}
```

TighteningControl

boltData

to get actual torque, angle and motor current. This Request is also possible if no tightening program is running.

Request:

```
{
"cmdID":"3",
"comp":"TighteningControl",
"fct":"boltData"
}
```

Response:

```
{
"retOfCmdID": "3",
"counter": 802826,
"timeStamp": {
        "tv_sec": 10199,
        "tv_nsec": 553994618
},
"boltTorque": 0.012451,
"boltAngle100": 4387.336838,
"motorCurrent": -0.100708
}
```

counter is a running number. If counter is a odd number following data are inconsistent.

timeStamp seconds and nano seconds. Seconds is only a 16 bit unsigned value. Be careful to handle the overflow.

boltTorque [Nm] the current torque on the bolt

boltAngle100 [angular degree] this value is in hardware a 16 bit signed integer and starts with value 0 on bootup and counts the incremental encoder with positive increments for clockwise- and negative increments for counter clockwise direction. The gear ratio is included in this value. This value will over-/underflow at about +- 15000 degree. This means if you start a tightening program with 3600 degree (10 rev.) at an actual boltAngle100 with 14000 degree you will not see 17600 after tightening. The value you will see is something like -2600 (minus).

motorCurrent [Ampere] in range of +- 85 A. Because of rounding errors the motor current is not exactly 0 in stand still.

gyroData

to get gyroscope's data. This Request is also possible if no tightening program is running.

Request:

```
{
"cmdID":"16",
"comp":"TighteningControl",
"fct":"gyroData"
}
```

Response:

```
{
        "retOfCmdID":  "16",
        "counter":     977536,
        "timeStamp":   {
                "tv_sec":      12117,
                "tv_nsec":     279279849
        },
        "gyAngleDx":   -1,
        "gyAngleDy":   5,
        "gyAngleDz":   20,
        "gyAccelDx":   -1033,
        "gyAccelDy":   94,
        "gyAccelDz":   -35
}
```

counter is a running number. If counter is a odd number following data are inconsistent.

timeStamp seconds and nano seconds. Seconds is only a 16 bit unsigned value. Be careful to handle the overflow.

gyAngleD… raw angle velocity data in x, y and z direction

gyAccelD… raw (gravity) acceleration data in x, y and z direction. At rest this is the gravity acceleration.

setSpeed

set current tightening speed (only effective if a tightening program is running)

Request:

```
{
"cmdID":"17",
"comp":"TighteningControl",
"fct":"setSpeed",
"rpmSpeed":60.0
}
```

rpmSpeed in rotations per minute. From +-14 to +-759 rpm. With negative values the direction is counter clockwise.

Response:

```
{
        "retOfCmdID":  "17",
        "returned":    0
}
```

returned 0=OK

switchesState

get current status of Nexo's buttons

Request:

```
{
"cmdID":"18",
"comp":"TighteningControl",
"fct":"switchesState"
}
```

Response

```
{
        "retOfCmdID":  "4",
        "counter":     1424650,
        "timeStamp":   {
                "tv_sec":      1557392721,
                "tv_nsec":     533749000
        },
        "lrSwitch":    1,
        "startBtn":    0,
        "keyLeft":     0,
        "keyA": 0,
        "keyRight":    0
}
```

counter is a running number. If counter is a odd number following data are inconsistent.

timeStamp seconds and nano seconds. Seconds since 1.Jan.1970.

lrSwitch status of direction switch.
-1 = counter clockwise direction,
0 = off,
+1 = clockwise direction

startBtn [%] pressed. Range from 0 to 100. With about 25% the program will start. Values lower then 25% will stop the tightening program.

keyLeft status of the left display key
0 = key released
1 = key press (rising edge)
2 = key is pressed

keyA status of the middle display key – see keyLeft

keyRight status of the right display key – see keyLeft

subscribeButtons

to get any changes of Nexo's buttons automatically

Request:

```
{
"cmdID":"125",
"comp":"TighteningControl",
"fct":"subscribeButtons"
}
```

Response:

```
{
"retOfCmdID": "125",
"rslt": true
}
```

The automatical sent button state is e.g.:

```
{
"retOfCmdID": "buttonEvent",
"counter": 652178,
"timeStamp": {
  "tv_sec": 1557308312,
  "tv_nsec": 753041231
  },
"lrSwitch": 1,
"startBtn": 0,
"keyLeft": 0,
"keyA": 0,
"keyRight": 0
}
```

unsubscribeButtons

to stop automatical send of switchesState.

Request:

```
{
"cmdID":"126",
"comp":"TighteningControl",
"fct":"unsubscribeButtons"
}
```

Response:

```
{
"retOfCmdID":"126",
"rslt":true
}
```

setLED

set the LED. This function is only available if no tightening program is running!

Request:

```
{
"cmdID":"20",
"comp":"TighteningControl",
"fct":"setLED",
"ledRed":250,
"ledGreen":120,
"ledBlue":60
}
```

Response:

```
{
"retOfCmdID": "20",
"returned": 0
}
```

returned Number 0=OK, negative values indicates an error.

tpRunningState

get current running tightening program status

Request:

```
{
"cmdID":"19",
"comp":"TighteningControl",
"fct":"tpRunningState"
}
```

Response:

```
{
        "retOfCmdID":  "19",
        "counter":     8,
        "timeStamp":   {
                "tv_sec":      1557147055,
                "tv_nsec":     703981153
        },
        "selectedProg":        1,
        "activeStep":  1,
        "stepType":    1,
        "activeCategory":      3,
        "programmedSpeed":     400,
        "caTrqThreshold":      0,
        "caTfAngleActive":     true,
        "caTfAngleValue":      3600,
        "caTfTorqueActive":    true,
        "caTfTorqueValue":     1,
        "caTfTimeActive":      false,
        "caTfTimeValue":       0,
        "caMfAngleMinActive":  true,
        "caMfAngleMinValue":   3599,
        "caTotalAngle":        0,
        "caStepAngle": 0,
        "caMfAngleMaxActive":  true,
        "caMfAngleMaxValue":   4000,
        "caMfAngleMaxSwitching":      true,
        "caMfTorqueMinActive":true,
        "caMfTorqueMinValue":  -1,
        "caMfTorqueMinSwitching":     false,
        "caTorque":    0,
        "caMfTorqueMaxActive":true,
        "caMfTorqueMaxValue":  2,
        "caMfTorqueMaxSwitching":     true,
        "caMfMaxTimeValue":    10,
        "caStepTime":  0,
        "caTotalTime": 0
}
```

counter is a running number. If counter is a odd number following data are inconsistent.

timeStamp seconds and nano seconds. Seconds since 1.Jan.1970.

selectedProg the number of the selected tightening program

activeStep current active step in the tightening program
-1 = program is not running
0 = start step
1..n = program step

stepType type of the step
0 = start step
1 = standard tightening step
2 = end step

activeCategory 0 = unnamed
1 = tightening
2 = prelim. Torque
3 = end torque
4 = loosen
5 = rework
6..10 = function A..E
11 = start

programmed speed in rpm. A negative value indicates counter clockwise rotation.

caTrqThreshold in programmer's unit (e.g. Nm). If current active torque is greater then this value caStepAngle will count.

caTfAngleActive true if target function Angle is active

caTfAngleValue [degree] value of the target function Angle if caTfAngleActive is true

caTfTorqueActive true if target function Torque is active

caTfTorqueValue [programmer's unit] value of the target function Torque if caTfTorqueActive is true

caTfTimeActive true if target function Time is active

caTfTimeValue [seconds] value of the target function Time if caTfTimeActive is true

caMfAngleMinActive true if monitor function AngleMin is active

caMfAngleMinValue [degree] value of the monitor function AngleMin if caTfAngleMinActive is true

caMfAngleMaxActive true if monitor function AngleMax is active

caMfAngleMaxValue [degree] value of the monitor function AngleMax if caTfAngleMaxActive is true

caMfAngleMaxSwitching true if monitor function AngleMax is over the complete step active
false if monitor function AngleMax is only evaluated at the end of the step

caMfTorqueMinActive true if monitor function TorqueMin is active

caMfTorqueMinValue [programmer's unit] value of the monitor function TorqueMin if caTfTorqueMinActive is true

caMfTorqueMinSwitching true if monitor function TorqueMin is over the complete step active
false if monitor function TorqueMin is only evaluated at the end of the step

caMfTorqueMaxActive true if monitor function TorqueMax is active

caMfTorqueMaxValue [programmer's unit] value of the monitor function TorqueMax if caTfTorqueMaxActive is true

caMfTorqueMaxSwitching	true if monitor function TorqueMax is over the complete step active false if monitor function TorqueMax is only evaluated at the end of the step
caMfMaxTimeValue	[seconds] value of the monitor function MaxTime
caStepTime	[seconds] current active time duration for this step
caTotalTime	[seconds] current active time duration for the complete tightening program
caStepAngle	[degree] current active angle value in this step. Remains 0 until caTrqThreshold is reached.
caTotalAngle	[degree] current active angle value over the complete tightening program
caTorque	[programmer's unit] current active torque value

subscribeProgram

sends tpRunningState automatical if a program is started or changed the program step.

Request:

```
{
"cmdID":"30",
"comp":"TighteningControl",
"fct":"subscribeProgram"
}
```

Response:

```
{
"retOfCmdID": "30",
"rslt": true
}
```

The automatically send program running status looks e.g.

```
{
"retOfCmdID": "programEvent",
"counter": 8,
"timeStamp": {
    "tv_sec": 1557388767,
    "tv_nsec": 963533000
  },
"selectedProg": 0,
"activeStep": 1,

:::

}
```

See tpRunningState

unsubscribeProgram

to stop automatical send of tpRunningState.

Request:

```
{
"cmdID":"1234",
"comp":"TighteningControl",
"fct":"unsubscribeProgram"
}
```

Response:

```
{
"retOfCmdID":"1234",
"rslt":true
}
```

motorLtTemperature

get current temperatures of motor and LT cooler.

Request:

```
{
"cmdID":"19",
"comp":"TighteningControl",
"fct":" motorLtTemperature"
}
```

Response

```
{
        "retOfCmdID":  "19",
        "motorTemperature":    39.600000,
        "ltCoolerTemperature":39.100000
}
```

motorTemperature in grad celcius.

ltCoolerTemperature in grad celcius.

ToolBattery

batteryStatus

to get the actual status and capacity of the battery.

Request:

```
{
"cmdID":"1235",
"comp":"ToolBattery",
"fct":"batteryStatus"
}
```

Response:

```
{
        "retOfCmdID":  "1235",
        "batteryPower":        66,
        "batteryStatus":       3
}
```

batteryPower	Number	[%] remaining capacity
batteryStatus	Number	0: The battery pack slide-in module is not inserted. 1: The battery level is critical (system is shut down) 2: The battery level is insufficient for any more tightening jobs. 3: The battery level is okay. 4: The battery was reinserted. 5: The battery level warning level is reached.