| **CopleyControl Class Commands** | | | |
| --- | --- | --- | --- |
| Commands | Input Type | Output Type | **Description** |
| Init | DevVoid | DevVoid | init the device |
| Move | DevVoid | DevVoid | trigger the motor to move. |
| SendCommandResult | DevString | DevString | write a command and get the result of this command from the motor controller. |
| State | DevVoid | State | if the motor is in motion, the State is “MOVING”;  if the motor is stationary, the State is “ON”;  if the motor is out of power, the State is “OFF”; |
| Status | DevVoid | DevString | if the motor is in motion, the State is “Status is MOVING”;  if the motor is stationary, the State is “Status is ON”;  if the motor is out of power, the State is “Status is OFF”; |
| Write | DevString | DevVoid | write ASCII message to the controller directly |
| CheckMove | DevVoid | DevLong | the function returns 1, if the motor is moving.Otherwise returns 0. |
| CompleteMove | DevVoid | DevLong | waits for the move to be finished, does the backlash, if FlagBacklash |
| GetPosition | DevVoid | DevDouble | get the defined position |
| GetRegister | DevVoid | DevLong | returns the controller register contents, hardware |
| SetRegister | DevLong | DevVoid | loads the motor register with a value, the hardware (to be tested) |
| ResetMotor | DevVoid | DevLong | reset the motor |
| StartMove | DevVoid | DevLong | a motor with FlagMotorReady is started, the function returns immediately |
| StopMove | DevVoid | DevLong | stop a movement immediately. (to be tested) |
| MoveToCwLimit | DevVoid | DevLong | Moves the motor until the CW limit is reached (positive step direction). Software limits are ignored. StopMove works. |
| MoveToCcwLimit | DevVoid | DevLong | Moves the motor until the CCW limit is reached (negative step direction). Software limits are ignored. StopMove works. |

| **CopleyControl Class Attributes** | | | | |
| --- | --- | --- | --- | --- |
| Attributes | **R/W type** |  | Data type | Value, Range, Unit | **Description** |
| Acceleration | READ\_WRITE | DevDouble | Units: 10 counts/s2 | the acceleration |
| Deceleration | READ\_WRITE | DevDouble | Units: 10 counts/s2 | the deceleration |
| NodeID | READ | DevShort | Range: 0-127 | the CAN node address |
| Port | READ | DevString | Default value: “/dev/ttyS0” | the serial port |
| Position | READ\_WRITE | DevDouble | Units: Counts | the relative position |
| Velocity | READ\_WRITE | DevDouble | Units: 0.1 counts/s | the velocity |
| State | READ | State |  | device state.  if the motor is in motion, the State is “MOVING”;  if the motor is stationary, the State is “STANDBY”;  if the motor is out of power, the State is “OFF”; |
| Status | READ | DevString |  | device status.  if the motor is in motion, the State is “Status is MOVING”;  if the motor is stationary, the State is “Status is ON”;  if the motor is out of power, the State is “Status is OFF”; |
| FlagBacklash | READ |  | DevLong |  | It shows always 0 if the motor is not moving or a backlash will not be added at the end of the started movement.  (Internally set to 1 if a backlash compensation will be performed at the end of the current movement.) |
| FlagMotorReady | READ |  | DevLong |  | It shows always 0 if the motor is not moving.  If the motor is in motion, returns 1. |