Rotary Actuator with Absolute Encoder - Interface and Integration

Contents

Revision History

Author	Date	Description
Jon Voiculescu	03/31/2021	Draft 1

Overview

The Rotary Actuator includes inside the housing all the necessary circuitry (logic and driver) for position control using standard RS422 (RS485 duplex on 4 wires) serial communication (19200 baud, 8, N, 1). The actuator is equipped with an absolute encoder mounted on the output shaft. The position information (30 bits) is retained between power cycles.

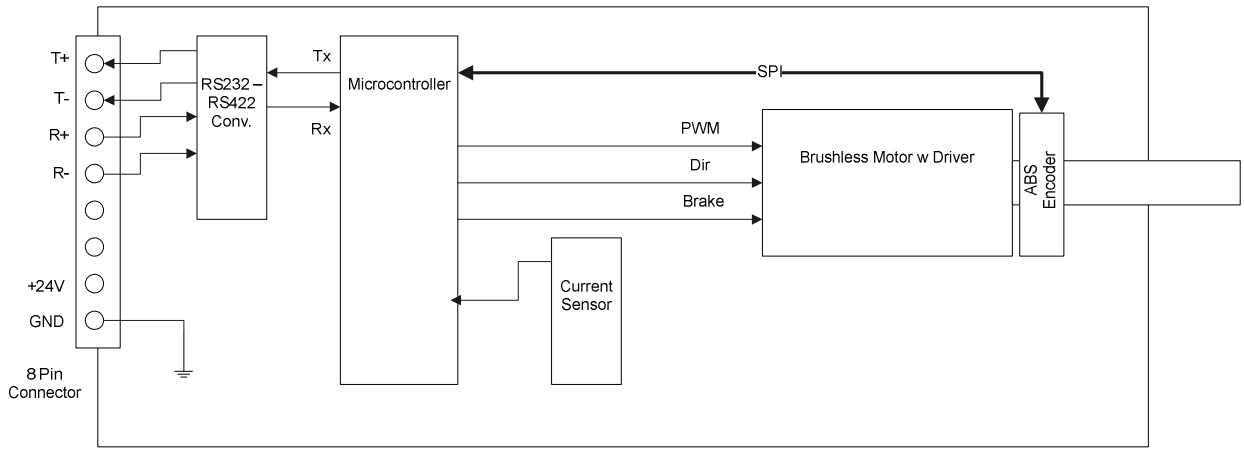
The multi-turn rotary encoder provides a 14 bits counter on single turn (16,383 counts per turn), and an additional 16 bits turns counter.

Two virtual switches are made available (software switches) for limiting the actuator travel distance between user configurable limits.

By default, the actuator is set to broadcast a status message as a sequence of 17 bytes with a frequency of 10HZ. The frequency of the status message is configurable and stored in the internal EEPROM. Other user configurable parameters are available as well. Refer to the Configuration Settings section of this document

The actuator can function in a continuous spin mode (jog) as well as position control with a precision of 0.1 deg at the output shaft.

Block Diagram



Power connections are not shown on the diagram above.

Status Message

The status message is a sequence of 17 bytes, and has the following structure:

0	1	2	3	4	5 - 9	10	11	12	13	14	15	16
Type = 135	**RPM Sign**	**RPM LB**	**RPM HB**	**Pos. Sign**	**Pos. LB … Pos. HB**	**Amps LB**	**Amps HB**	**Flags**	**Error LB**	**Error HB**	**Check Sum**	**Term = 255**

Only the first and last byte have the most significant bit set. All other bytes have the most significant bit zero

Byte	Description
0	Message Type. Values: - 135 (0x87) – Message is a status message (144 (0x90) – Message is a configuration message, see configuration section)
1	Speed sign. Values: 0: RPM value (bytes 2 and 3) is negative (shaft rotation is CCW) 1: RPM value (bytes 2 and 3) is positive (shaft rotation is CW)
2, 3	Speed Absolute Value in encoder counts per 10 ms. RPM Low Byte: bits 0 to 6 of the Speed value, RPM High Byte: bits 7 to 13 of the Speed value. To calculate the actual speed multiply the resulting value by 100, then divide by number of counts per unit. (For deg., the “counts per unit” is 16384/360.) The result obtained will be in degrees/sec.
4	Position sign. Values: 0: Position value (bytes 5 to 9) is negative 1: Position value (bytes 5 to 9) is positive
5, 6, 7, 8, 9	Current Position Absolute Value in encoder units. Byte 5: Current Position Low Byte (byte 0) : bits 0 to 6 of the position value, Byte 6: Current Position Byte 1: bits 7 to 13 of the position value, … Byte 9: Current Position High Byte (byte 4) : bits 28, 29 of the position value (see below this table for calculation of the position value and transformation in length units)
10,11	Current Consumption Low Byte: bits 0 to 6 of the current value, Current Consumption High Byte: bits 7 to 13 of the current value. The two bytes combine into a number from 0 to 1023 representing a current from 0A to 10A. The current sensor (Allegro’s ACS724) accepts a unidirectional current input up to 10 A, and outputs a proportional analog voltage (400 mV/A) that measures 500 mV when the input current is zero. The formula to calculate the current is (N-102)/82 where N is the number resulting from the 2 bytes The output is noisy – may require averaging / filtering
12	Flags. Each bit in the flags byte has the following signification: - Bit 0 (least significant) – 1 = Brake Off, 0 = Brake On - Bit 1 – 1 = Position Reached, 0 = Position Not reached - Bit 2 – Always 1 – no meaning. - Bit 3 – 0 = “Encoder Warning”. Position is valid, but some parameters may be at limit. It is normal for this flag to be raised shortly at power up. - Bit 4 – 1 = “Whiplash”. A request was received to change direction without slowing down or

	stopping first. The actuator will decelerate and stop. - Bit 5 – 1 = Limit switch 1 reached (min allowed position), 0 = L1 not reached - Bit 6 – 1 = Limit switch 2 reached (max allowed position), 0 = L2 not reached
13,14	Error Number Low Byte: bits 0 to 6 of the error value, Error Number High Byte: bits 7 to 13 of the error value. The bytes combine into a number from 0 to 2047. Refer to the Errors section for details
15	Check Sum. XOR of all the bytes up to the checksum position. The most significant bit is set to zero after calculation
16	Termination Byte. Always 255 (0xFF)

Example calculation of the actual value represented by a pair of bytes:

```
Value = (High Byte << 7) + Low Byte, or
Value = Low Byte + 128 * High Byte
```

Example calculation of the actual value represented by 5 bytes (position – bytes 5 to 9, representing the 5 bytes as b0, b1, b2, b3, b4 with b0 being the least significant):

```
counterValue = (b4 << 28) + (b3 << 21) + (b2 << 14) + (b1 << 7) + b0, or
counterValue = b0 + 128 * b1 + (128 ^ 2) * b2 + (128 ^ 3) * b3 + (128 ^ 4) * b4
```

Example function for transforming the encoder value in measuring units (degrees):

```
'VB.NET
Private Function GetUnitsValue(ByVal counterValue As Integer) As Single
      Dim result As Single
      Dim countsPerUnit As Single = 16384 / 360
      result = Math.Round(counterValue / countsPerUnit, 4)
      Return result
End Function
```

The 17 bytes are sent by the actuator one byte at the time (not as a compact packet). On the receiving side more than 1 read event may be raised to receive the full sequence.

Note: Getting / Setting configuration parameters responds with different messages – refer to the section Get / Set Configuration Commands.

Configuration settings / commands affecting the status message:

- Talk Back Interval (configuration): sets the broadcast frequency. If set to zero the broadcast is stopped
- Enter / Exit Configuration (command): entering configuration mode stops the broadcast of the status message, exiting resumes broadcasting
- Get Status (command): forces a status response out of sequence. Useful if the Talk back Interval is set to zero

Commands

Commands are sent as binary bytes in packets, and have the following general structure:

Command ID	Parameter1	Parameter2	…	…	Parameter8	Check Sum	Termination Byte

Byte	Description
Command ID	Unique identifier for each command. The most significant bit is always set
Parameters 1 to 8	Specific for each command. The most significant bit is always zero
Check Sum	XOR of all the bytes up to the checksum position. The most significant bit must be set to zero after calculation
Termination Byte	Always 255 (0xFF)

- **Only the first and last bytes of the command have the most significant bit set. All other in between have the most significant bit zero**
- **Commands are variable in length. The termination byte must be the last one in a packet with no un-necessary bytes following.**

All commands take effect immediately. The previous command will be discarded even if execution is not complete and the new command will be executed. The only exception is when the new command will cause the actuator to change direction without stopping first. In this case the “whiplash” flag is raised and a “stop” command is generated internally before the new command is executed.

Unknown commands or commands with errors are ignored (errors are reported).

Except for the “Get Status”, commands do not generate a response unless the configuration “Talk Back Interval” (TBI) is set to zero. If TBI is zero, all commands will generate a Status response.

Configuration Get/Set commands always generate a specific response depending on Configuration ID.

Spin. Command (JOG) ID = 128 (0x80)

Commands the actuator to move continuously with a set speed and set direction:

	Command ID	**Duty**	**Direction**	**Check Sum**	**Termination**
Decimal	128	100	0		255
Hex	80	64	00		FF

Parameter	Description
Duty	Values: Dead Band value to 127 for zero to maximum speed
Direction	0 = Counter Clock Wise, 1 = Clock Wise

Example:
Decimal: 128, 50, 1, 51, 255 Hex: 0x80, 0x32, 0x51, 0xFF (Rotate CW with a speed of 50)

- **The actuator will keep moving until a "stop" command is issued or until the virtual switches are reached – even if communication is lost**

Configuration settings / commands affecting the Spin Command:

- Dead Band (configuration): If the duty cycle parameter is less than Dead Band value, the speed will be set to zero.
- Minimum / Maximum Virtual Switches: If required to move past a virtual switch value will raise an "Over Limit" error (will not move).

Go To Position. Command ID = 129 (0x81)

Commands the actuator to move to a defined position expressed in encoder counter value, using a defined speed.

	Command ID	**Absolute /Relative**	**Pos. Sign**	**Pos. Low Byte (P0)**	**P1 … P3**	**Pos. High Byte (P4)**	**Duty**	**Check Sum**	**Termination**
Byte	0	1	2	3	4,5,6	7	8	9	10
Decimal	129	1	1	0	0, 0, 0	0	20	21	255
Hex	81	01	01	00	00, 00, 00	00	14	15	FF

Parameter	Description
Absolute / Relative (1)	0 = Relative: Moves to a position calculated from the current position. (Increase / Decrease position by X encoder counts) 1 (or any value other than zero) = Absolute: Moves to specified position in relation to the origin. (Go to the X encoder counts position)
Sign (2)	0 = Negative. For relative position it will rotate CCW. 1 (or any value other than zero) = Positive. For relative position it will rotate CW. For Absolute positioning the sign should be always 1.
Position Bytes 0 to 4 (3) to (7)	Target position value in encoder counts. (P0 is the least significant byte.) Each byte is a 7 bits chunk of the full possible 30 bits value.
Duty	Values: Dead Band value to 127 for zero to maximum speed

Example calculation of the position bytes (encoder counts value):

```
Pos[3] = value & 0x7F                    //least significant byte
Pos[4] = (value >> 7) & 0x7F
Pos[5] = (value >> 14) & 0x7F
Pos[6] = (value >> 21) & 0x7F
Pos[7] = (value >> 28) & 0x7F            //most significant byte
```

Calculating the Encoder Counter Value from the actual units value (degrees):

```
'VB.NET
Private Function GetCounterValue(ByVal unitsValue As Single) As Integer
       Dim countsPerUnit As Single = 16384 / 360              'unit = deg
       Dim result As Integer
       result = Convert.ToInt32(unitsValue * countsPerUnit)
       Return result
End Function
```

Example:

Decimal: 129, 1, 1, 0, 0, 0, 0, 0, 20, 21, 255 Hex: 0x81, 0x01, 0x01, 0x00… 0x00, 0x14, 0x15, 0xFF

(Go To Zero with a speed of 20)

Configuration settings / commands affecting the Position Command:

- Minimum (Virtual Switch 1) (configuration): The actuator will not be allowed to rotate CCW past the minimum value. The command will be executed but the actuator will stop at the minimum allowed value. A command to rotate CCW when the minimum is already reached will raise a “Over Limit” error.
- Maximum (Virtual Switch 2) (configuration): The actuator will not be allowed to rotate CW past the maximum value. The command will be executed but the actuator will stop at the maximum allowed value. A command to expand when the maximum is already reached will raise a “Over Limit” error.
- Deceleration Space (configuration): Distance in the vicinity of the target at which the actuator slows down to avoid overshooting the target position.
- Deceleration Min. Duty (configuration): Reduced speed value in the vicinity of the target. If the speed specified in the command (parameter 8) is lower than Deceleration Min. Duty, Deceleration Min. Duty will be applied. Deceleration Min. Duty must be greater than Dead Band (or the actuator will stop when reaching the low speed angle).

Flags affecting the Position Command:

- Flag bit 1 – Position Reached. Information only. Set to “1” when command completes without interruption. The command can be interrupted by other commands issued before the position command completes.

Stop. Command ID = 131 (0x83)

Stops the actuator.

	Command ID	Any	Check Sum	Termination
Decimal	131	0	3	255
Hex	83	00	03	FF

No parameters (“Any” can be any value). The values above represent a good example.

No configuration settings affect this command.

Clear Errors. Command ID = 132 (0x84)

Clears the error flags.

	Command ID	Any	Check Sum	Termination
Decimal	132	0	4	255
Hex	84	00	04	FF

No parameters (“Any” can be any value). The values above represent a good example.

No configuration settings affect this command.

Get Status. Command ID = 135 (0x87)

Request a status message.

	Command ID	Any	Check Sum	Termination
Decimal	135	0	7	255
Hex	87	00	07	FF

No parameters (“Any” can be any value). The values above represent a good example. This command is useful mainly if the configuration setting “Talk Back Interval” is set to zero. This command doesn’t work in configuration mode. An “exit configuration mode” must be issued first.

Enter / Exit Configuration. Command ID = 134 (0x86)

Stops the Status Message broadcasting gets the actuator ready for changing configuration settings.

This command generates a response representing the configuration setting for Configuration ID = 0:
144,0,0,1,28,99,0,0,0,0,0,0, 0,0,0,110,255

- **Important: Make sure the actuator is stopped when you enter (start) configuration mode**

	Command ID	Start/Stop	Check Sum	Termination
Decimal	134	0	6	255
Hex	86	00	06	FF

Parameter Start / Stop:
= 1 Enters the Configuration Mode (Stops Broadcasting, responds as above)
= 0 Exits the Configuration Mode (Resumes Status Message Broadcasting)

Example:
134, 1, 7, 255 0x86, 0x01, 0x07, 0xFF Enter Configuration Mode

Configuration Commands and Settings

Configuration parameters values are stored on the internal EEPROM and the values are preserved between power cycles. Each configuration setting has a unique ID. All configuration settings get or set commands generate responses.

- **Important: Make sure the actuator is stopped when you enter (start) configuration mode.**
- **Do not perform configuration changes if a “Load Driven” error is reported.**

During EEPROM writes the internal interrupt routines are disabled. EEPROM write may fail if the actuator is moving.

Get / Set Configuration. Command ID = 144 (0x90)

Gets or Sets a specific configuration setting. This command generates response. All values passed or received are Encoder Counter values.

	Command ID	**Configuration ID**	**Get/Set**	**Value Low Byte (C0)**	**Value Bytes C1 – C3**	**Value High Byte (C4)**	**Check Sum**	**Termination**
Decimal	144	1	0	0	0	0	16	255
Hex	90	01	00	00	00	00	10	FF

Parameter	Description
Configuration ID	See table below for each configuration details
Get / Set	0 = Get. Returns the value of the specific configuration in the response 1 (or any value other than zero) = Set. Writes the specified value to EEPROM
Value (5 Bytes)	In chunks of 7 bits, LSB first. Refer to the status message section and position command details for examples of calculation. **All configuration settings are positive values.**

Response structure (17 Bytes) – applies the same structure and rules as the status message:

0	1	2	3	4 … 8	9 … 12	13	14	15	16
Type = 144	**Config. ID**	**Get / Set**	**1**	**Value (5 bytes, LSB first)**	**0**	**Error LB**	**Error HB**	**Check Sum**	**Term = 255**

Byte		Description
0		Response Type. 144 (0x90) for all configuration responses
1		Configuration ID. Unique identifier for each configuration setting
2		0: Response to a Get Command 1: Response to a Set Command
3		1 (One)
4 … 8		In chunks of 7 bits, LSB first. Refer to the status message section and position command details for examples of calculation. All configuration settings are positive values.
9 … 12		0 (Zero)
13,14		Error Number Low 7 bits, Error Number High 7 bits. Refer to the Errors section for details
15		Check Sum. calculated as XOR of all the bytes from 0 to 14 with the most significant bit set to zero after calculation
16		Termination byte. Always 255 (0xFF)

List of Configuration Settings:

Config. ID	Name	Description
0	Zero Offset	This is a constructive constant and is set at the assembly time. Used for the alignment of the zero angle position with the mounting holes.
1	Talk Back Interval	Sets the frequency (time interval between) for Status Messages broadcast. Values: 1 byte, number (0 – 127), representing the time interval between status messages in units of 10 ms. If = 0 or < 10, the broadcast is disabled. After each command a status message will be sent back. If => 10, broadcast is enabled and status messages will be sent at the specified interval Default Setting: 10 (100 ms = 10 Hz)
2	Acceleration Step / Dead Band	Sets the minimum value for duty (speed) as well as the increment for acceleration. The actuator cannot start from zero. The default setting has been determined with zero load. If you encounter frequent “Stall” errors at very low speed you may need to increase this setting. Values: 1 byte, number (0 – 127) If greater than “Deceleration Min. Duty” value, the actuator will stop before reaching target (at +/- “Deceleration Space”). Default Setting: 7
3	Deceleration Min. Duty	Sets the duty (speed) value for the final approach to the target position. Represents also the minimum speed for position control. Values: 1 byte, number (0 – 127) If less than Dead Band value, the actuator will stop before reaching target (at +/- “Deceleration Space”). Default Setting: 10
4	Deceleration Space	Sets the proximity to target position for speed reduction Values: 5 bytes, in Encoder Counter values, greater than zero. If = 0, there will be no speed reduction (**will deteriorate gear box and precision**) Default Setting: Approx. 1200 (13 Degrees at the shaft)
5	Minimum (Virtual SW1)	The actuator will not be allowed to rotate CCW past the minimum value. Speed will be reduced when the position is within deceleration space and the motion is CCW. Cannot be bigger than Maximum – will raise “Over Limit” error. Values: 5 bytes, in Encoder Counter values, greater than zero. Default Setting: 0
6	Maximum (Virtual SW2)	The actuator will not be allowed to rotate CW past the maximum value. Speed will be reduced when the position is within deceleration space and the motion is CW. Cannot be lower than Minimum or bigger than “Stroke” – will raise “Over Limit” error. Values: 5 bytes, in Encoder Counter values, greater than zero. Default Setting: Close to stroke value
7	Stroke	Maximum length of travel. This is a constructive constant and is set at the assembly time.

Error Messages.

Error flags are packed in two bytes and returned in each response on the position 13 and 14. The reported error applies to the last command sent. The significance of each bit in the error bytes is explained below:

Bit #	Error	Description
Byte 13, bit 0	Encoder Error	The reported position information is not valid. Remediation: Perform a power cycle. It is normal for this bit to come on shortly at power on.
Byte 13, bit 1	Unknown Command	The command sent is not recognized (refers to first byte of the command only) Result: The command is discarded
Byte 13, bit 2	Receiver Overflow	The internal buffer of the actuator receiver (2 bytes) has overflown. Result: Even if valid, the last command is probably incomplete and will not be executed. Remediation: Repeat the last command.
Byte 13, bit 3	Missing Termination Byte	The termination byte (0xFF) could not be found in the received command buffer. Result: The command will not be executed. If this is due to communication interference, other error flags may be raised as well. Remediation: Repeat the last command.
Byte 13, bit 4	Bad Checksum	The checksum calculated by the actuator does not match the checksum byte in the command packet. Result: The command will not be executed. If this is due to communication interference, other error flags may be raised as well. Remediation: Repeat the last command.
Byte 13, bit 5	Over Limit	A request was sent to the actuator to move past the limit switches values, or a configuration command attempts to set conflicting values for limit switches. Remediation: None
Byte 13, bit 6	Stalled	The actuator is required to move but it cannot (too much load or physical obstacle). If the condition persists for 2.4 seconds the actuator will issue a stop command on its own. The condition must persist for 0.5 seconds for the flag to be raised. Result: The actuator will stop if the stall persists for 2.4 seconds Remediation: Issue a “stop” command and attempt moving the actuator in the opposite direction. Verify the load and check for physical obstacles.
Byte 14, bit 0	Load Driven	The actuator moves while the current applicable command is “stop”. In case of inertial loads, it may be that the load is driving the actuator. Result: The current position will be updated. This is more an information flag, not an actual error. Remediation: None **Important: do not perform calibration or configuration changes if this flag is raised.**
Byte 14, Bit 1	Parameter Out of Bounds	One or more of the command parameters have the most significant bit set. Result: The command will not be executed. If this is due to communication interference, other error flags may be raised as well. Remediation: Repeat the last command.
Byte 14, Bit 2	Wrong Number of Parameters	Too many or few bytes between the command byte and the termination byte. Result: The command will not be executed. If this is due to communication interference, other error flags may be raised as well. Remediation: Verify logic or repeat the last command.

Byte 14, Bit 3	Bad Config. ID	A configuration command was issued for a configuration setting that is not in the list. Result: The command will not be executed. If this is due to communication interference, other error flags may be raised as well. Remediation: Verify logic or repeat the last command.