

# QuadQuad2Drv PCB Datasheet

4ch Motor Controller and Quadrature Decoder

---

## Description

The QuadQuad2Drv microprocessor provides closed loop control of up to four DC motors. Incremental quadrature encoders are decoded to provide position, velocity and metadata. The device communicates as SPI slave and data can be either polled or streamed. Home/index/edge inputs are provided.

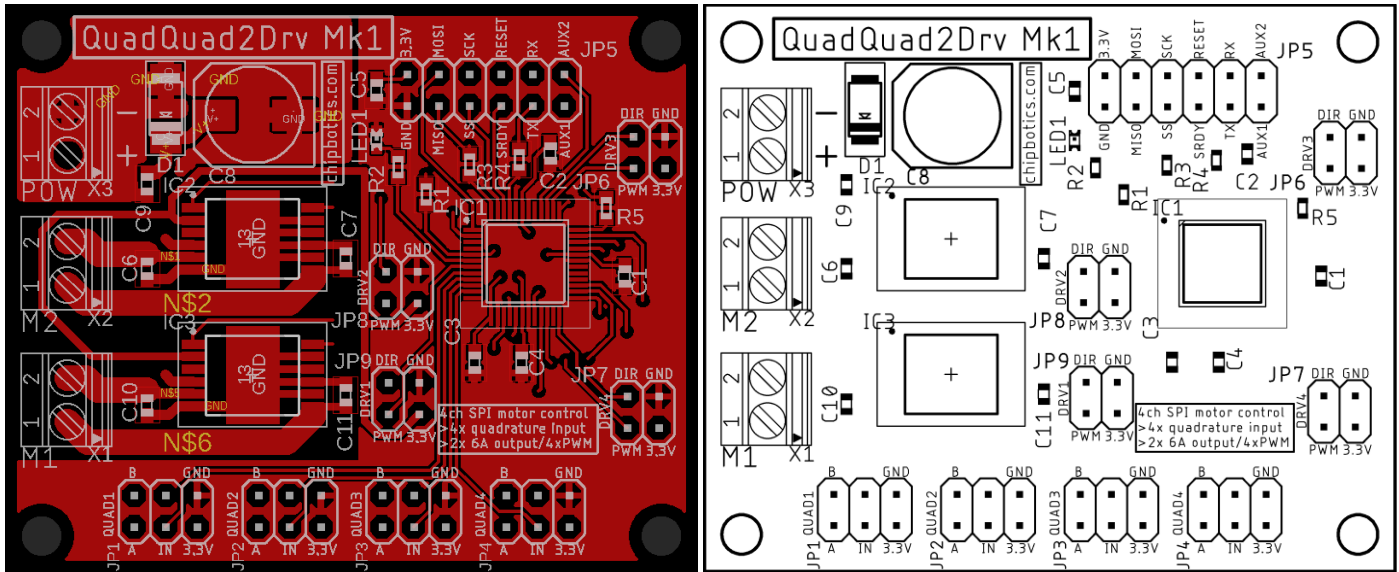
## Features

- Four motor outputs (digital PWM at 11kHz and direction)
- Two 6A drivers allow two motors to be connected directly
- Four quadrature feedback channels
- Closed loop PID control:
  - Four modes: Off, Power, Position PID, Velocity PID
  - Position and velocity ramp commands
  - Power, velocity and acceleration limiting
  - Loop rate of 500Hz
- Feedback data provided:
  - Position (8, 16 or 32-bit)
  - Velocity
  - Metadata: status, stream timing
- Rated up to 50,000 quadrature transitions/s with velocity resolution of 4 transitions/s
- Home/index/edge input for each encoder
- Operating voltage 3.3V with 5V tolerant SPI inputs
- Bootloader for firmware updates via UART serial
- Ideal for controlling N20 gearmotors, such as the Pololu HP motors with extended shaft and quadrature encoder board
- Arduino library and demo code downloadable
- PIC library available on request

# Contents

Description .....	1
Features .....	1
Physical Connections.....	3
Electrical Characteristics .....	4
Bootloader & Firmware Updates .....	5
Protocol.....	6
SPI Interface .....	6
Packet Format .....	6
Packets .....	6
1 - Get Version .....	6
2 - Get Binary Motion Data .....	6
3 - Binary Stream Data .....	8
4 - Set Stream Period .....	8
5 - Get Stream Period.....	8
6 - Set Data Mask .....	9
7 - Get Data Mask.....	9
8 - Set Stream Config.....	10
9 - Get Stream Config.....	10
10 - Set Position .....	10
11 - Get Position.....	11
12 - Set History Dimensions .....	11
13 - Get History Dimensions .....	12
14 – Set Input Mode.....	12
16 – Set Control Mode .....	14
18 – Set Motor Power .....	14
20 – Set Position Control Target .....	15
22 – Set Velocity Control Target .....	15
24 – Set Power Limit .....	15
26 – Set Velocity Limit.....	16
28 – Set Acceleration Limit .....	16
30 – Set Position Control PID Constants .....	17
32 – Set Velocity Control PID Constants .....	17
34 – Position Ramp .....	18
36 – Velocity Ramp .....	19
Worked Examples .....	20
Read Version Numbers .....	20

# Physical Connections



The QuadQuad2Drv PCB features several connectors:

## Digital Communications Block

Name	Interface	Function
GND (VSS)	Power	Digital Ground
3.3V (VDD)	Power	Digital Power 3.3V
MISO	Output	SPI Data Out (Connect to Data In / MISO on master)
MOSI	Input (5V tolerant)	SPI Data In (Connect to Data Out / MOSI on master)
SS	Input (5V tolerant)	SPI Slave Select Input (Active low, has pull-up)
SCK	Input (5V tolerant)	SPI Serial Clock Input
SRDY	Output	SPI Data Ready Output (See section <i>SPI Interface</i> )
MCLR	Input	Reset Pin Input (Active low, has pull-up)
TX	Output	UART Receive and Transmit at 115,200bps. (Used by bootloader)
RX	Input	
AUX1	Output	Used to access bootloader (Short AUX1 and AUX2 and reset/power up to start bootloader)
AUX2	Input (5V tolerant)	

## Quadrature Inputs

There are four identical quadrature input blocks labeled “QUAD1” to “QUAD4”.

Name	Function
A, B	Quadrature Inputs A and B
IN	Home/index/edge Inputs (Optional, polarity is software configurable)
GND, 3.3V	Power supply to quadrature encoder

## Digital Motor Outputs

There are four identical digital motor output blocks labeled “DRV1” to “DRV4”. DRV1 and DRV2 serve as inputs to power drivers for M1 and M2 onboard motor power drivers.

Name	Function
PWM	Pulse-Width Modulation Output

DIR	Motor Direction Output
GND, 3.3V	Power Supply to External Power Driver

### Motor Power Connectors

Name	Function
M1	Motor 1 Power Output (6A)
M2	Motor 2 Power Output (6A)
POW	Motor Power Supply (max 35V) for Motor Power Outputs

### Electrical Characteristics

- Digital operating voltage 3.3V. Absolute maximum 3.0V – 4.0V.
- SPI inputs and AUX2 are 5V tolerant (MOSI, SS, SCK, AUX2). Absolute maximum 3.0V – 5.5V.
- Quadrature input blocks are not 5V tolerant!
- Motor power voltage maximum 35V.
- Motor power output maximum 6A each.
- Digital input pins are Schmitt triggers with low/high thresholds of 0.2 *VDD* and 0.8 *VDD*.
- Digital output pin low/high voltages are max 0.42V and min 2.4V.

## Bootloader & Firmware Updates

The QuadQuad2Drv firmware can be updated via UART serial using the bootloader. A common USB-to-serial cable can be used. Connect the serial cable as follows:

Cable Wire	Board Pin	Note
GND	GND	Connect to any of the GND pins on the board
TX	RX	
RX	TX	

Take care to use only a serial cable with 3.3V interface and power the board only with 3.3V. A serial cable with 3.3V power output is convenient as they can also be used to power the board (such as the FTDI TTL-232RG-VSW3V3-WE or TTL-232RG-VREG3V3-WE) by connecting the cable 3.3V VCC to the 3.3V board pin. The common FTDI TTL-232R-3V3 can be used, but since this cable has a 5V output, the board must be powered some other way.

The bootloader can be accessed by shorting AUX1 and AUX2 with a jumper and resetting or powering up the board.

Use a serial terminal that supports line delays such as the popular Tera Term. For Tera Term, go to *Setup->Serial port* and set the transmit delay to 50 msec/line. Set the baud rate to 115,200 bps.

When the terminal is set up, the serial cable is connected, the AUX pins are shorted and the board is powered up or reset, the bootloader will print a header line specifically including the word "bootloader". For Tera Term, the firmware file can be dragged and dropped on the terminal window and the bootloader will confirm each line. If the upload is successful, the jumper can be removed and the board reset or power cycled or type "reset" in the terminal window.

# Protocol

## SPI Interface

The SPI master interface must be configured for SPI mode 1. We recommend clocking SPI up to 500kHz.

The QuadQuad2Drv SPI interface includes a handshake line from the slave to master, called *Serial Ready* (SRDY), which is set high by the slave to indicate that a new data packet is available.

The following sequence should be followed for a typical packet query:

1. Master waits until slave asserts *Serial Ready* (SRDY) low.
2. Master asserts *Slave Select* (SS) low.
3. Master reads out one entire packet.
4. Master may leave *Slave Select* (SS) low.

Stream packets are sent at a steady rate without requiring a query packet. When a stream packet becomes available, the slave asserts SRDY and the stream packet can be read.

## Packet Format

Applicable interface protocol version: v1.

Size	U8	U8	U8	Variable	U8
Field	STX	Packet Size	Packet ID	Payload	Checksum

*STX* is ASCII character 2.

*Packet Size* is the size of the entire packet, in bytes.

*Checksum* is the 8-bit checksum of the entire packet.

## Packets

The following section contains details of the packets that can be communicated, specifically, the *Payload* field. The documented fields below are read left to right first, then row by row.

Packet ID	Description / Payload		
1 - Get Version	Read firmware and protocol version numbers.		
	Send	None	
	Reply	Size	U8
		Field	Return Code
Field		Firmware Version Major	
Reply	Size	U8	
	Field	Firmware Version Minor	
	Field	Protocol Version	
2 - Get Binary Motion Data	Read motion data. Each quadrature channel and field in the reply payload, except for <i>Return Code</i> , is optional and configurable using the 6 - <i>Set Data Mask</i> and 8 - <i>Set Stream Config</i> packets. Channels and fields that are deselected will simply be omitted (zero bytes). Alternatively, this data can be streamed using packet 4 - <i>Set Stream Period</i> .		
	Send	None	

Size	U8	Variable	Variable
Field	Return Code	[Channel 1 Data]	[Channel 2 Data]
Size	Variable	Variable	
Field	[Channel 3 Data]	[Channel 4 Data]	

**Channel Data:**

Size	I8/I16/I32	I32	U8
Field	[Position]	[Velocity]	[Status]

**Position:** Number of quadrature transitions forward/backward. There are four transitions per detent. The number of bits used to represent *Position* can be configured using packet 6 - *Set Data Mask* and defaults to 32-bits. When less than 32-bit position is specified, the lower 8- or 16 bits will simply be retrieved and wrapping will occur on overflow. When relative position mode is enabled using packet 6 - *Set Data Mask*, *Position* will contain the change in position since the last position read.

**Velocity:** Rate at which *Position* is changing, measured as:  
 Transitions Per Second = *Velocity*.

There are four transitions per detent. Velocity is unaffected by home/index inputs.

**Status:**

Bit	7	6	5..4
Field	Glitch	Overspeed	Reserved
Bit	3	2	1..0
Field	Input Active	Input Active Accumulator	Input Trigger Accumulator

Reply

**Glitch:** This bit is set if the two quadrature lines A and B have made an invalid transition. This may be caused by noise on the lines or if quadrature velocity exceeds the rated maximum velocity. *Position* and *Velocity* data may be inaccurate if this bit is set.

**Overspeed:** Quadrature velocity has exceeded rated maximum velocity and *Position* and *Velocity* data may be inaccurate.

**Input Active:** The input is currently active (evaluated after polarity setting applied).

**Input Active Accumulator:** The input has been active at least once since the last time status was transmitted (evaluated after polarity setting applied).

**Input Trigger Accumulator:** If non-zero, the input has been triggered at least once since the last time status was transmitted. The Position counter has been set as configured using packet 14 – *Set Input Mode*, or the index/home/edge position has been recorded. This differs from the *Input Active* flags in that a trigger requires additional conditions. For example, an index trigger also requires the quadrature A and B lines both to be 0 and an edge trigger is only set once when the input transitions from inactive to active.

The two bits indicate which edge has been triggered in HOME and EDGE input modes. When input mode is set to INDEX, the Positive Trigger will always be used. Also see packet 14 – *Set Input Mode*.

Bit	1	0
Field	Negative Edge	Positive Edge

3 - Binary Stream Data	Stream packet with motion data. Each quadrature channel and field in the reply payload, except for <i>Return Code</i> , is optional and configurable using the 6 - <i>Set Data Mask</i> and 8 - <i>Set Stream Config</i> packets. Fields that are deselected will simply be omitted (zero bytes). Use packet 4 - <i>Set Stream Period</i> to set up streaming. Also see section "SPI Interface" for how to use the <i>Serial Ready (SRDY)</i> handshake line.																
	Send	This packet must not be sent by the master device.															
	Reply	<table border="1"> <tr> <td>Size</td> <td>U16</td> <td>U8</td> <td>Variable</td> </tr> <tr> <td>Field</td> <td>[Stream Period Timing]</td> <td>[Stream Periods Elapsed]</td> <td>[Channel 1 Data]</td> </tr> <tr> <td>Size</td> <td>Variable</td> <td>Variable</td> <td>Variable</td> </tr> <tr> <td>Field</td> <td>[Channel 2 Data]</td> <td>[Channel 3 Data]</td> <td>[Channel 4 Data]</td> </tr> </table> <p><b>Stream Period Timing:</b> Time elapsed since start of calculation of previous stream packet until the current stream packet became available for transmission, measured as: <math>\text{Stream Period Timing in Microseconds} = \text{Stream Period Timing} \times 40.96</math>. In short, this number measures how old the stream data is and is typically the configured <i>Stream Period</i> plus a small processing overhead. Example: If you set <i>Stream Period</i> = 100 and measure <i>Stream Period Timing</i> = 108, the processing overhead since capturing the quadrature data for the current packet and making it available for transmission is <math>8 \times 40.96\text{ms}</math>. If the master fails to read an entire stream packet before the next period, a new stream packet will not be provided until the master reads out the packet. <i>Stream Period Timing</i> excludes time for any missed stream periods. <i>Stream Period Timing</i> will clip at 0xFFFF.</p> <p><b>Stream Periods Elapsed:</b> Number of <i>Stream Periods</i> elapsed since the last successful stream packet transmission. This will normally be 1, unless <i>Stream Period</i> is set too small such that the master cannot read the packet in time. <i>Stream Periods Elapsed</i> will clip at 0xFF.</p> <p><b>Channel Data:</b> See packet 2 - <i>Get Binary Motion Data</i>.</p>	Size	U16	U8	Variable	Field	[Stream Period Timing]	[Stream Periods Elapsed]	[Channel 1 Data]	Size	Variable	Variable	Variable	Field	[Channel 2 Data]	[Channel 3 Data]
Size	U16	U8	Variable														
Field	[Stream Period Timing]	[Stream Periods Elapsed]	[Channel 1 Data]														
Size	Variable	Variable	Variable														
Field	[Channel 2 Data]	[Channel 3 Data]	[Channel 4 Data]														
4 - Set Stream Period	Set period at which motion data is streamed to the master. Alternatively, motion data can be polled using packet 2 - <i>Get Binary Motion Data</i> .																
	Send	<table border="1"> <tr> <td>Size</td> <td>U16</td> </tr> <tr> <td>Field</td> <td>Stream Period</td> </tr> </table> <p><b>Stream Period:</b> Time between stream packets, measured as: <math>\text{Stream Period in Microseconds} = \text{Stream Period} \times 40.96</math>. Set <i>Stream Period</i> = 0 to disable the stream. Set <i>Stream Period</i> = 1 to receive stream packets at maximum rate. In reality, stream packets cannot be sent at very high rates such as period = 1. The actual rate is affected by the amount of processing required, such as number of fields in the stream, SPI clock speed and the master response time. Also see <i>Stream Period Timing</i> in packet 3 - <i>Binary Stream Data</i>. Default at start-up: 0 (Disabled)</p>	Size	U16	Field	Stream Period											
	Size	U16															
Field	Stream Period																
Reply	<table border="1"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code												
Size	U8																
Field	Return Code																
5 - Get Stream Period	Read period at which motion data is streamed to the master.																
	Send	None															
	Reply	<table border="1"> <tr> <td>Size</td> <td>U8</td> <td>U16</td> </tr> <tr> <td>Field</td> <td>Return Code</td> <td>Stream Period</td> </tr> </table> <p><b>Stream Period:</b> See packet 4 - <i>Set Stream Period</i>.</p>	Size	U8	U16	Field	Return Code	Stream Period									
Size	U8	U16															
Field	Return Code	Stream Period															



6 - Set Data Mask	<p>Set which channels and fields are to be included in motion data (see packet 2 - <i>Get Binary Motion Data</i> and 3 - <i>Binary Stream Data</i>).</p> <table border="1" data-bbox="464 181 1378 255"> <tr> <th>Size</th> <td>U8</td> <td>U8</td> <td>U8</td> <td>U8</td> </tr> <tr> <th>Field</th> <td>Chan Mask 1</td> <td>Data Mask 1</td> <td>[Chan Mask...]</td> <td>[Data Mask...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and <i>Data Mask</i> pairs can be concatenated to set multiple masks to multiple channels. If any mask conflicts occur, any latter bit value will override the former.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Data Mask</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1" data-bbox="464 508 1355 582"> <tr> <th>Bit</th> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <th>Field</th> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Data Mask:</b> Bit mask selecting what data to include for channels selected by <i>Channel Mask</i>.</p> <table border="1" data-bbox="464 689 1386 799"> <tr> <th>Bit</th> <td>7..6</td> <td>5</td> <td>4</td> <td>3</td> <td>2</td> <td>1..0</td> </tr> <tr> <th>Field</th> <td>Unused</td> <td>Status</td> <td>Reserved</td> <td>Velocity</td> <td>Position Relative</td> <td>Position Size</td> </tr> </table> <p><b>Position Size:</b> Select how many bits will be used to represent position value.</p> <table border="1" data-bbox="464 871 951 1057"> <thead> <tr> <th>Position Size</th> <th>Number of Bits</th> </tr> </thead> <tbody> <tr> <td>0x00</td> <td>Position omitted</td> </tr> <tr> <td>0x01</td> <td>8-bit signed integer</td> </tr> <tr> <td>0x02</td> <td>16-bit signed integer</td> </tr> <tr> <td>0x03</td> <td>32-bit signed integer</td> </tr> </tbody> </table> <p>Default at start-up: 0x03 (32-bit signed integer)</p> <p><b>Position Relative:</b> Set this bit to enable relative position mode. This mode will cause position values to represent the change in position since the last read position value (via either packet 2 - <i>Get Binary Motion Data</i> or 3 - <i>Binary Stream Data</i>) instead of absolute position value. Absolute position value can still be read using packet 11 - <i>Get Position</i>.</p> <p>Default at start-up: 0 (Disabled)</p> <p><b>Velocity:</b> Set this bit to include velocity data.</p> <p>Default at start-up: 1 (Enabled)</p> <p><b>Status:</b> Set this bit to include status data. See packet 2 - <i>Get Binary Motion Data</i>.</p> <p>Default at start-up: 0 (Disabled)</p> <table border="1" data-bbox="464 1581 759 1657"> <tr> <th>Size</th> <td>U8</td> </tr> <tr> <th>Field</th> <td>Return Code</td> </tr> </table>	Size	U8	U8	U8	U8	Field	Chan Mask 1	Data Mask 1	[Chan Mask...]	[Data Mask...]	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1	Bit	7..6	5	4	3	2	1..0	Field	Unused	Status	Reserved	Velocity	Position Relative	Position Size	Position Size	Number of Bits	0x00	Position omitted	0x01	8-bit signed integer	0x02	16-bit signed integer	0x03	32-bit signed integer	Size	U8	Field	Return Code
Size	U8	U8	U8	U8																																															
Field	Chan Mask 1	Data Mask 1	[Chan Mask...]	[Data Mask...]																																															
Bit	7..4	3	2	1	0																																														
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																																														
Bit	7..6	5	4	3	2	1..0																																													
Field	Unused	Status	Reserved	Velocity	Position Relative	Position Size																																													
Position Size	Number of Bits																																																		
0x00	Position omitted																																																		
0x01	8-bit signed integer																																																		
0x02	16-bit signed integer																																																		
0x03	32-bit signed integer																																																		
Size	U8																																																		
Field	Return Code																																																		
7 - Get Data Mask	<p>Read which channels and fields are to be included in motion data (packets 2 - <i>Get Binary Motion Data</i> and 3 - <i>Binary Stream Data</i>).</p> <table border="1" data-bbox="464 1749 1490 1800"> <tr> <td>Send</td> <td>None</td> </tr> </table> <table border="1" data-bbox="464 1809 1347 1962"> <tr> <th>Size</th> <td>U8</td> <td>U8</td> <td>U8</td> </tr> <tr> <th>Field</th> <td>Return Code</td> <td>Chan 1 Data Mask</td> <td>Chan 2 Data Mask</td> </tr> <tr> <th>Size</th> <td>U8</td> <td>U8</td> <td></td> </tr> <tr> <th>Field</th> <td>Chan 3 Data Mask</td> <td>Chan 4 Data Mask</td> <td></td> </tr> </table> <p>The <i>Data Mask</i> for each channel is returned.</p>	Send	None	Size	U8	U8	U8	Field	Return Code	Chan 1 Data Mask	Chan 2 Data Mask	Size	U8	U8		Field	Chan 3 Data Mask	Chan 4 Data Mask																																	
Send	None																																																		
Size	U8	U8	U8																																																
Field	Return Code	Chan 1 Data Mask	Chan 2 Data Mask																																																
Size	U8	U8																																																	
Field	Chan 3 Data Mask	Chan 4 Data Mask																																																	

		<b>Data Mask:</b> See packet 6 - <i>Set Data Mask</i> .																																													
8 - Set Stream Config	Set which additional fields to include in stream data.																																														
	Send	<table border="1"> <tr> <td>Size</td> <td colspan="3">U8</td> </tr> <tr> <td>Field</td> <td colspan="3">Stream Config Mask</td> </tr> </table> <p><b>Stream Config Mask:</b></p> <table border="1"> <tr> <td>Bit</td> <td>7..2</td> <td>1</td> <td colspan="2">0</td> </tr> <tr> <td>Field</td> <td>Reserved</td> <td>Stream Periods Elapsed</td> <td colspan="2">Stream Period Timing</td> </tr> </table> <p><b>Stream Period Timing:</b> See packets 3 - <i>Binary Stream Data</i> and 4 - <i>Set Stream Period</i>. Default at start-up: 0 (Disabled)</p> <p><b>Stream Periods Elapsed:</b> See packets 3 - <i>Binary Stream Data</i> and 4 - <i>Set Stream Period</i>. Default at start-up: 0 (Disabled)</p>	Size	U8			Field	Stream Config Mask			Bit	7..2	1	0		Field	Reserved	Stream Periods Elapsed	Stream Period Timing																												
Size		U8																																													
Field	Stream Config Mask																																														
Bit	7..2	1	0																																												
Field	Reserved	Stream Periods Elapsed	Stream Period Timing																																												
	Reply	<table border="1"> <tr> <td>Size</td> <td colspan="3">U8</td> </tr> <tr> <td>Field</td> <td colspan="3">Return Code</td> </tr> </table>	Size	U8			Field	Return Code																																							
Size	U8																																														
Field	Return Code																																														
9 - Get Stream Config	Read which additional fields to include in stream data.																																														
	Send	None																																													
	Reply	<table border="1"> <tr> <td>Size</td> <td>U8</td> <td colspan="2">U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> <td colspan="2">Stream Config Mask</td> </tr> </table> <p><b>Stream Config Mask:</b> See packet 8 - <i>Set Stream Config</i>.</p>	Size	U8	U8		Field	Return Code	Stream Config Mask																																						
Size	U8	U8																																													
Field	Return Code	Stream Config Mask																																													
10 - Set Position	Set absolute position value.																																														
	Send	<table border="1"> <tr> <td>Size</td> <td>U8</td> <td colspan="2">18/116/132</td> </tr> <tr> <td>Field</td> <td>Channel &amp; Size Mask 1</td> <td colspan="2">Position 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td colspan="2">18/116/132</td> </tr> <tr> <td>Field</td> <td>[Channel &amp; Size Mask...]</td> <td colspan="2">[Position...]</td> </tr> </table> <p>Any number of <i>Channel &amp; Size Mask</i> and <i>Position</i> pairs can be concatenated to set different position values for multiple channels.</p> <p><b>Channel &amp; Size Mask:</b> Select which channels to write <i>Position</i> value to and the format of the <i>Position</i> value. Multiple channels can be selected simultaneously.</p> <table border="1"> <tr> <td>Bit</td> <td>7..6</td> <td>5..4</td> <td colspan="2">3..0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Position Size</td> <td colspan="2">Channel Mask</td> </tr> </table> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Position</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1"> <tr> <td>Bit</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Position Size:</b> Select the size of the <i>Position</i> value following. The whole 32-bit position value will be overwritten even if an 8-bit or 16-bit value is specified.</p> <table border="1"> <tr> <td>Position Size</td> <td>Number of Bits</td> </tr> <tr> <td>0x00</td> <td><i>Position</i> omitted and assumed zero</td> </tr> <tr> <td>0x01</td> <td>8-bit signed integer</td> </tr> <tr> <td>0x02</td> <td>16-bit signed integer</td> </tr> <tr> <td>0x03</td> <td>32-bit signed integer</td> </tr> </table>	Size	U8	18/116/132		Field	Channel & Size Mask 1	Position 1		Size	U8	18/116/132		Field	[Channel & Size Mask...]	[Position...]		Bit	7..6	5..4	3..0		Field	Unused	Position Size	Channel Mask		Bit	3	2	1	0	Field	Channel 4	Channel 3	Channel 2	Channel 1	Position Size	Number of Bits	0x00	<i>Position</i> omitted and assumed zero	0x01	8-bit signed integer	0x02	16-bit signed integer	0x03
Size		U8	18/116/132																																												
Field	Channel & Size Mask 1	Position 1																																													
Size	U8	18/116/132																																													
Field	[Channel & Size Mask...]	[Position...]																																													
Bit	7..6	5..4	3..0																																												
Field	Unused	Position Size	Channel Mask																																												
Bit	3	2	1	0																																											
Field	Channel 4	Channel 3	Channel 2	Channel 1																																											
Position Size	Number of Bits																																														
0x00	<i>Position</i> omitted and assumed zero																																														
0x01	8-bit signed integer																																														
0x02	16-bit signed integer																																														
0x03	32-bit signed integer																																														

		<b>Position:</b> The <i>Position</i> value to be written. The size must match the size specified in <i>Position Size</i> . If <i>Position Size</i> = 0x00, <i>Position</i> must be omitted and is assumed <i>Position</i> = 0.																																
	Reply	<table border="1"> <tr> <td>Size</td> <td colspan="2">U8</td> <td colspan="4"></td> </tr> <tr> <td>Field</td> <td colspan="2">Return Code</td> <td colspan="4"></td> </tr> </table>						Size	U8						Field	Return Code																		
Size	U8																																	
Field	Return Code																																	
11 - Get Position	Read absolute position value.																																	
	Send	Three different parameter formats are possible and will determine the contents and format of the reply. The format is identified by the size of the payload.																																
		<b>Size 0:</b> An empty payload will request the position values in the sizes previously configured using <i>Position Size</i> in packet 6 - <i>Set Data Mask</i> , or omitted if the size was set to 0.																																
		<table border="1"> <tr> <td>Size</td> <td colspan="2">Zero</td> <td colspan="4"></td> </tr> <tr> <td>Field</td> <td colspan="2">None</td> <td colspan="4"></td> </tr> </table>						Size	Zero						Field	None																		
Size		Zero																																
Field		None																																
		<b>Size 1:</b> A single byte will specify a channel mask. The position values of the channels selected in <i>Channel Mask</i> will be returned in the reply.																																
		<table border="1"> <tr> <td>Size</td> <td colspan="2">U8</td> <td colspan="4"></td> </tr> <tr> <td>Field</td> <td colspan="2">Channel Mask</td> <td colspan="4"></td> </tr> </table>						Size	U8						Field	Channel Mask																		
Size		U8																																
Field		Channel Mask																																
		<b>Size 4:</b> Four bytes will specify the size/format of the position value, for each channel, to return in the reply.																																
	<table border="1"> <tr> <td>Size</td> <td colspan="2">U8</td> <td colspan="2">U8</td> <td colspan="2"></td> </tr> <tr> <td>Field</td> <td colspan="2">Channel 1 Position Size</td> <td colspan="2">Channel 2 Position Size</td> <td colspan="2"></td> </tr> <tr> <td>Size</td> <td colspan="2">U8</td> <td colspan="2">U8</td> <td colspan="2"></td> </tr> <tr> <td>Field</td> <td colspan="2">Channel 3 Position Size</td> <td colspan="2">Channel 4 Position Size</td> <td colspan="2"></td> </tr> </table>						Size	U8		U8				Field	Channel 1 Position Size		Channel 2 Position Size				Size	U8		U8				Field	Channel 3 Position Size		Channel 4 Position Size			
Size	U8		U8																															
Field	Channel 1 Position Size		Channel 2 Position Size																															
Size	U8		U8																															
Field	Channel 3 Position Size		Channel 4 Position Size																															
	<b>Channel Mask:</b>																																	
	<table border="1"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table>						Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																
Bit	7..4	3	2	1	0																													
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																													
	<b>Channel Position Size:</b>																																	
	<table border="1"> <tr> <td>Position Size</td> <td>Number of Bits</td> </tr> <tr> <td>0x00</td> <td>Position omitted</td> </tr> <tr> <td>0x01</td> <td>8-bit signed integer</td> </tr> <tr> <td>0x02</td> <td>16-bit signed integer</td> </tr> <tr> <td>0x03</td> <td>32-bit signed integer</td> </tr> </table>						Position Size	Number of Bits	0x00	Position omitted	0x01	8-bit signed integer	0x02	16-bit signed integer	0x03	32-bit signed integer																		
Position Size	Number of Bits																																	
0x00	Position omitted																																	
0x01	8-bit signed integer																																	
0x02	16-bit signed integer																																	
0x03	32-bit signed integer																																	
	Reply	<table border="1"> <tr> <td>Size</td> <td>U8</td> <td>I8/I16/I32</td> <td>I8/I16/I32</td> <td>I8/I16/I32</td> <td>I8/I16/I32</td> </tr> <tr> <td>Field</td> <td>Return Code</td> <td>[Position 1]</td> <td>[Position 2]</td> <td>[Position 3]</td> <td>[Position 4]</td> </tr> </table>					Size	U8	I8/I16/I32	I8/I16/I32	I8/I16/I32	I8/I16/I32	Field	Return Code	[Position 1]	[Position 2]	[Position 3]	[Position 4]																
Size	U8	I8/I16/I32	I8/I16/I32	I8/I16/I32	I8/I16/I32																													
Field	Return Code	[Position 1]	[Position 2]	[Position 3]	[Position 4]																													
		The presence and size/format of each <i>Position</i> value is determined by the specified parameters.																																
12 - Set History Dimensions	Set maximum history length and averaging time for velocity calculation purposes.																																	
	Send	<table border="1"> <tr> <td>Size</td> <td>U8</td> <td colspan="2">U8</td> <td colspan="2"></td> </tr> <tr> <td>Field</td> <td>History Length</td> <td colspan="2">Maximum Averaging Time in Bits</td> <td colspan="2"></td> </tr> </table>					Size	U8	U8				Field	History Length	Maximum Averaging Time in Bits																			
Size		U8	U8																															
Field	History Length	Maximum Averaging Time in Bits																																
		<b>History Length:</b> The maximum number of quadrature capture events that will be averaged. A capture event occurs on every fourth quadrature transition, i.e. once per detent. Larger values will produce more accurate velocity averages, but will be slower to respond to changes in velocity. Range: 2 - 127																																

		<p>Default at start-up: 31</p> <p><b>Maximum Averaging Time in Bits:</b> The maximum time over which quadrature capture events are averaged, specified in bits, where:  Maximum Averaging Time = <math>2^{\text{Maximum Averaging Time in Bits}}</math> and  Maximum Averaging Time in Microseconds = <i>Maximum Averaging Time</i> x 0.64</p> <table border="1" data-bbox="464 327 1294 1104"> <thead> <tr> <th>Maximum Averaging Time in Bits</th> <th>Maximum Averaging Time</th> <th>Maximum Averaging Time in Milliseconds</th> </tr> </thead> <tbody> <tr><td>14</td><td>16384</td><td>10.5</td></tr> <tr><td>15</td><td>32768</td><td>21.0</td></tr> <tr><td>16</td><td>65536</td><td>41.9</td></tr> <tr><td>17</td><td>131072</td><td>83.9</td></tr> <tr><td>18</td><td>262144</td><td>167.8</td></tr> <tr><td>19</td><td>524288</td><td>335.5</td></tr> <tr><td>20</td><td>1048576</td><td>671.1</td></tr> <tr><td>21</td><td>2097152</td><td>1,342.2</td></tr> <tr><td>22</td><td>4194304</td><td>2,684.4</td></tr> <tr><td>23</td><td>8388608</td><td>5,368.7</td></tr> <tr><td>24</td><td>16777216</td><td>10,737.4</td></tr> <tr><td>25</td><td>33554432</td><td>21,474.8</td></tr> <tr><td>26</td><td>67108864</td><td>42,949.7</td></tr> <tr><td>27</td><td>134217728</td><td>85,899.3</td></tr> <tr><td>28</td><td>268435456</td><td>171,798.7</td></tr> <tr><td>29</td><td>536870912</td><td>343,597.4</td></tr> <tr><td>30</td><td>1073741824</td><td>687,194.8</td></tr> <tr><td>31</td><td>2147483648</td><td>1,374,389.5</td></tr> <tr><td>32</td><td>4294967296</td><td>2,748,779.1</td></tr> </tbody> </table> <p>At lower velocities, the total duration of <i>History Length</i> quadrature capture events increases and thus the lower the minimum detectable velocity will be, but the slower average velocity will wind down when motion is significantly slowed down or stopped, since there will be physically less transitions to detect. Larger values will allow lower velocities to be detected.  Range: 14 - 32  Default at start-up: 20</p> <p>At very low velocities, when <i>History Length</i> quadrature capture events exceeds <i>Maximum Averaging Time</i>, less than <i>History Length</i> capture events will be used for averaging, but allowing for faster responses and longer total averaging times instead.</p>				Maximum Averaging Time in Bits	Maximum Averaging Time	Maximum Averaging Time in Milliseconds	14	16384	10.5	15	32768	21.0	16	65536	41.9	17	131072	83.9	18	262144	167.8	19	524288	335.5	20	1048576	671.1	21	2097152	1,342.2	22	4194304	2,684.4	23	8388608	5,368.7	24	16777216	10,737.4	25	33554432	21,474.8	26	67108864	42,949.7	27	134217728	85,899.3	28	268435456	171,798.7	29	536870912	343,597.4	30	1073741824	687,194.8	31	2147483648	1,374,389.5	32	4294967296	2,748,779.1
Maximum Averaging Time in Bits	Maximum Averaging Time	Maximum Averaging Time in Milliseconds																																																															
14	16384	10.5																																																															
15	32768	21.0																																																															
16	65536	41.9																																																															
17	131072	83.9																																																															
18	262144	167.8																																																															
19	524288	335.5																																																															
20	1048576	671.1																																																															
21	2097152	1,342.2																																																															
22	4194304	2,684.4																																																															
23	8388608	5,368.7																																																															
24	16777216	10,737.4																																																															
25	33554432	21,474.8																																																															
26	67108864	42,949.7																																																															
27	134217728	85,899.3																																																															
28	268435456	171,798.7																																																															
29	536870912	343,597.4																																																															
30	1073741824	687,194.8																																																															
31	2147483648	1,374,389.5																																																															
32	4294967296	2,748,779.1																																																															
13 - Get History Dimensions	<p>Reply</p> <table border="1" data-bbox="464 1518 762 1597"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																																																												
Size	U8																																																																
Field	Return Code																																																																
14 – Set Input Mode	<p>Send</p>	<p>Read maximum history length and averaging time for velocity calculation purposes.</p> <p>Send: None</p> <p>Reply</p> <table border="1" data-bbox="464 1713 1417 1792"> <tr> <td>Size</td> <td>U8</td> <td>U8</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> <td>History Length</td> <td>Maximum Averaging Time in Bits</td> </tr> </table> <p>Set the function of quadrature input pins as home / index / edge / disabled. The status of the input can be read in the &lt;Status&gt; field using packet 2 - <i>Get Binary Motion Data</i> or packet 3 - <i>Binary Stream Data</i>.</p> <p>Send</p> <table border="1" data-bbox="464 1921 1449 2078"> <tr> <td>Size</td> <td>U8</td> <td>U8</td> <td>U16</td> <td>I32</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>Input Configuration 1</td> <td>Spacing 1</td> <td>Position 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>U8</td> <td>U16</td> <td>I32</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[Input Configuration...]</td> <td>[Spacing...]</td> <td>[Position...]</td> </tr> </table>				Size	U8	U8	U8	Field	Return Code	History Length	Maximum Averaging Time in Bits	Size	U8	U8	U16	I32	Field	Channel Mask 1	Input Configuration 1	Spacing 1	Position 1	Size	U8	U8	U16	I32	Field	[Channel Mask...]	[Input Configuration...]	[Spacing...]	[Position...]																																
Size	U8	U8	U8																																																														
Field	Return Code	History Length	Maximum Averaging Time in Bits																																																														
Size	U8	U8	U16	I32																																																													
Field	Channel Mask 1	Input Configuration 1	Spacing 1	Position 1																																																													
Size	U8	U8	U16	I32																																																													
Field	[Channel Mask...]	[Input Configuration...]	[Spacing...]	[Position...]																																																													

Any number of *Channel Mask*, *Input Configuration* and *Position* pairs can be concatenated to set different input modes for multiple channels.

**Channel Mask:** Bit mask selecting which channels *Input Mode* and *Position* will be applied to. Multiple channels can be selected by adding masks.

Bit	7..4	3	2	1	0
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1

**Input Configuration:**

Bit	7..6	5..4	3	2	1..0
Field	Unused	Edge	Input Polarity	Reserved	Input Mode

**Edge:** Select how to behave when an input trigger occurs. Either the position counter is set to *<Position>* specified in this packet, or the current position counter is just recorded. The module records the extreme positions reached before an input is activated. In HOME and EDGE modes, the position will only be set on the specified positive/negative end. If *<Spacing>* is set to 0, a single end-stop is assumed and *<Edge>* will specify on which end the end-stop is. The direction is irrelevant in INDEX mode.

Input Polarity	Meaning
0x00	Record position only. If <i>&lt;spacing&gt;</i> =0, this is a positive end-stop.
0x01	Record position only. If <i>&lt;spacing&gt;</i> =0, this is a negative end-stop.
0x02	Set position counter on positive end trigger
0x03	Set position counter on negative end trigger

**Input Polarity:** Select whether the input pin will be active when low or high.

Input Polarity	Meaning
0x00	Input is active when low
0x01	Input is active when high

**Input Mode:**

Input Mode	Meaning	Description
0x00	Disabled	Signal on input has no effect.
0x01	Home mode	While input is active (see <i>Input Polarity</i> ), <i>Position</i> will be written into relevant channel position value if so configured. This is typically used on linear actuators where a button or infrared sensor acts as input to indicate that the end stop has been reached.
0x02	Index mode	While input is active (see <i>Input Polarity</i> ) and quadrature input A = B = 0, <i>Position</i> will be written into relevant channel position value if so configured. This is typically used with an index signal output from a quadrature encoder.
0x03	Edge mode	When an input transition from inactive to active occurs, <i>Position</i> will be written into relevant channel position value if so configured. This is ideal for end-stop sensors.

**Spacing:** Specifies a hysteresis threshold when arranged such that the input is triggered on either positive/negative end. This can be when using two separate sensors on each end, where either can set the input, or when dealing with circular motion where a single sensor can be set in either direction. The hysteresis is typically

		<p>a small value used to prevent triggering on the wrong end by requiring the positive end to be this distance away from the negative end and vice versa.</p> <p>Further, this is used to specify whether there is one or two end-stops. A value of 0 indicates one end-stop and any other values indicates two.</p> <p><b>Position:</b> The <i>Position</i> value to be written. If <i>Input Mode</i> is set to <i>Disabled</i>, <i>Position</i> must be omitted.</p>																																							
	Reply	<table border="1"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																																			
Size	U8																																								
Field	Return Code																																								
16 – Set Control Mode	Set the motor control mode.	<table border="1"> <tr> <td>Size</td> <td>U8</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>Control Mode 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[Control Mode...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and <i>Control Mode</i> pairs can be concatenated to set different motor control modes for multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Control Mode</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Control Mode:</b></p> <table border="1"> <thead> <tr> <th>Input Mode</th> <th>Meaning</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0x00</td> <td>Off</td> <td>Motor output will be set to zero / off.</td> </tr> <tr> <td>0x01</td> <td>Power</td> <td>Motor output will be set to the power value specified using packet 18 – Set Motor Power.</td> </tr> <tr> <td>0x02</td> <td>Position PID Control</td> <td>Motor output will be controlled using closed loop PID to maintain a position target. See position control related commands.</td> </tr> <tr> <td>0x03</td> <td>Velocity PID Control</td> <td>Motor output will be controlled using closed loop PID to maintain a velocity target. See velocity control related commands.</td> </tr> </tbody> </table>	Size	U8	U8	Field	Channel Mask 1	Control Mode 1	Size	U8	U8	Field	[Channel Mask...]	[Control Mode...]	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1	Input Mode	Meaning	Description	0x00	Off	Motor output will be set to zero / off.	0x01	Power	Motor output will be set to the power value specified using packet 18 – Set Motor Power.	0x02	Position PID Control	Motor output will be controlled using closed loop PID to maintain a position target. See position control related commands.	0x03	Velocity PID Control	Motor output will be controlled using closed loop PID to maintain a velocity target. See velocity control related commands.
Size	U8	U8																																							
Field	Channel Mask 1	Control Mode 1																																							
Size	U8	U8																																							
Field	[Channel Mask...]	[Control Mode...]																																							
Bit	7..4	3	2	1	0																																				
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																																				
Input Mode	Meaning	Description																																							
0x00	Off	Motor output will be set to zero / off.																																							
0x01	Power	Motor output will be set to the power value specified using packet 18 – Set Motor Power.																																							
0x02	Position PID Control	Motor output will be controlled using closed loop PID to maintain a position target. See position control related commands.																																							
0x03	Velocity PID Control	Motor output will be controlled using closed loop PID to maintain a velocity target. See velocity control related commands.																																							
	Reply	<table border="1"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																																			
Size	U8																																								
Field	Return Code																																								
18 – Set Motor Power	Set the motor output power. Control Mode must be set to <i>Power</i> using packet 16 – Set Control Mode.	<table border="1"> <tr> <td>Size</td> <td>U8</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>Power 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>l16</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[Power...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and <i>Power</i> pairs can be concatenated to set different motor output powers for multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Power</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table>	Size	U8	U8	Field	Channel Mask 1	Power 1	Size	U8	l16	Field	[Channel Mask...]	[Power...]	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1															
Size	U8	U8																																							
Field	Channel Mask 1	Power 1																																							
Size	U8	l16																																							
Field	[Channel Mask...]	[Power...]																																							
Bit	7..4	3	2	1	0																																				
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																																				

	<p><b>Power:</b> Power to apply to the motor output. The sign determines the direction. Range: -8,191 - 8,191 Default at start-up: 0</p>																												
Reply	<table border="1"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																								
Size	U8																												
Field	Return Code																												
20 – Set Position Control Target	<p>Set the motor position control target. Control Mode must be set to <i>Position PID Control</i> using packet 16 – <i>Set Control Mode</i>.</p> <table border="1"> <tr> <td>Size</td> <td>U8</td> <td>I32</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>Position Target 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>I32</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[Position Target...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and <i>Control Mode</i> pairs can be concatenated to set different position targets for multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Position Target</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Position Target:</b> Position target the closed loop PID will aim to achieve.</p> <table border="1"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	I32	Field	Channel Mask 1	Position Target 1	Size	U8	I32	Field	[Channel Mask...]	[Position Target...]	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1	Size	U8	Field	Return Code
Size	U8	I32																											
Field	Channel Mask 1	Position Target 1																											
Size	U8	I32																											
Field	[Channel Mask...]	[Position Target...]																											
Bit	7..4	3	2	1	0																								
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																								
Size	U8																												
Field	Return Code																												
22 – Set Velocity Control Target	<p>Set the motor velocity control target. Control Mode must be set to <i>Velocity PID Control</i> using packet 16 – <i>Set Control Mode</i>.</p> <table border="1"> <tr> <td>Size</td> <td>U8</td> <td>I32</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>Velocity Target 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>I32</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[Velocity Target...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and <i>Control Mode</i> pairs can be concatenated to set different velocity targets for multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Velocity Target</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Velocity Target:</b> Velocity target the closed loop PID will aim to achieve, specified as: <i>Velocity Target</i> = &lt;transitions/s&gt;. Range: -100,000 shl 12 - 100,000 shl 12</p> <table border="1"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	I32	Field	Channel Mask 1	Velocity Target 1	Size	U8	I32	Field	[Channel Mask...]	[Velocity Target...]	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1	Size	U8	Field	Return Code
Size	U8	I32																											
Field	Channel Mask 1	Velocity Target 1																											
Size	U8	I32																											
Field	[Channel Mask...]	[Velocity Target...]																											
Bit	7..4	3	2	1	0																								
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																								
Size	U8																												
Field	Return Code																												
24 – Set Power Limit	<p>Set the motor power limit. This limit will be applied regardless of control mode.</p> <table border="1"> <tr> <td>Size</td> <td>U8</td> <td>U16</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>Power Limit 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>U16</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[Power Limit...]</td> </tr> </table>	Size	U8	U16	Field	Channel Mask 1	Power Limit 1	Size	U8	U16	Field	[Channel Mask...]	[Power Limit...]																
Size	U8	U16																											
Field	Channel Mask 1	Power Limit 1																											
Size	U8	U16																											
Field	[Channel Mask...]	[Power Limit...]																											

		<p>Any number of <i>Channel Mask</i> and <i>Power Limit</i> pairs can be concatenated to set different power limits for multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Power Limit</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1" data-bbox="464 286 1355 365"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Power Limit:</b> Value that output power will be limited to. Range: 0 - 8,191</p>	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1												
Bit	7..4	3	2	1	0																					
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																					
26 – Set Velocity Limit	Reply	<table border="1" data-bbox="464 495 759 573"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																				
Size	U8																									
Field	Return Code																									
28 – Set Acceleration Limit	Send	<p>Set the motor acceleration limit. This limit will be applied in both <i>Position PID Control</i> and <i>Velocity PID Control</i> modes. Additionally, in <i>Position PID Control</i> mode, the motor will decelerate at this rate in advance, as it approaches the position target, to prevent it from overshooting.</p> <p><b>Note:</b> This limit is not guaranteed as it depends on PID configuration and real-world interaction.</p> <table border="1" data-bbox="464 775 1197 925"> <tr> <td>Size</td> <td>U8</td> <td>U32</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>Velocity Limit 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>U32</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[Velocity Limit...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and <i>Velocity Limit</i> pairs can be concatenated to set different velocity limits for multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Velocity Limit</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1" data-bbox="464 1137 1355 1216"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Velocity Limit:</b> Value that control velocity will be limited to, specified as: <i>Velocity Limit</i> = &lt;transitions/s&gt; shl 12. The extra bits can be used to specify a fractional velocity. Such low velocities are not normally achievable in speed control mode and primarily apply to <i>Position PID Control</i> mode. Range: 0 - 100,000 shl 12</p>	Size	U8	U32	Field	Channel Mask 1	Velocity Limit 1	Size	U8	U32	Field	[Channel Mask...]	[Velocity Limit...]	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1
Size	U8	U32																								
Field	Channel Mask 1	Velocity Limit 1																								
Size	U8	U32																								
Field	[Channel Mask...]	[Velocity Limit...]																								
Bit	7..4	3	2	1	0																					
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																					
	Reply	<table border="1" data-bbox="464 1487 759 1565"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																				
Size	U8																									
Field	Return Code																									



	<p>Any number of <i>Channel Mask</i> and <i>Acceleration Limit</i> pairs can be concatenated to set different acceleration limits for multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>Acceleration Limit</i> will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1" data-bbox="464 286 1355 362"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Acceleration Limit:</b> Value that control acceleration will be limited to, specified as:  <i>Acceleration Limit</i> = &lt;transitions/s/s&gt; shl 12.  The extra bits can be used to specify a fractional acceleration. A value of zero disables acceleration limiting.  Range: 0 - 1000,000 shl 12</p>	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																								
Bit	7..4	3	2	1	0																																
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																																
	<p>Reply</p> <table border="1" data-bbox="464 600 759 676"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																																
Size	U8																																				
Field	Return Code																																				
<p>30 – Set Position Control PID Constants</p>	<p>Set the PID constants for <i>Position PID Control</i> mode.</p> <table border="1" data-bbox="464 734 1386 889"> <tr> <td>Size</td> <td>U8</td> <td>U16</td> <td>U16</td> <td>U16</td> <td>U16</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>P1</td> <td>I1</td> <td>D1</td> <td>Windup Limit 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>U16</td> <td>U16</td> <td>U16</td> <td>U16</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[P...]</td> <td>[I...]</td> <td>[D...]</td> <td>[Windup Limit ...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and <i>PID</i> pairs can be concatenated to set different PID constants for multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting which channels <i>PID</i> constants will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1" data-bbox="464 1104 1355 1180"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>P:</b> Proportional Constant.  Default: 100</p> <p><b>I:</b> Integration Constant.  Default: 500</p> <p><b>D:</b> Differential Constant.  Default: 500</p> <p><b>Windup Limit:</b> Integration windup limit.  Range: 0 - 8,191  Default: 600</p>	Size	U8	U16	U16	U16	U16	Field	Channel Mask 1	P1	I1	D1	Windup Limit 1	Size	U8	U16	U16	U16	U16	Field	[Channel Mask...]	[P...]	[I...]	[D...]	[Windup Limit ...]	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1
Size	U8	U16	U16	U16	U16																																
Field	Channel Mask 1	P1	I1	D1	Windup Limit 1																																
Size	U8	U16	U16	U16	U16																																
Field	[Channel Mask...]	[P...]	[I...]	[D...]	[Windup Limit ...]																																
Bit	7..4	3	2	1	0																																
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																																
	<p>Reply</p> <table border="1" data-bbox="464 1666 759 1742"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																																
Size	U8																																				
Field	Return Code																																				
<p>32 – Set Velocity Control PID Constants</p>	<p>Set the PID constants for <i>Velocity PID Control</i> mode.</p> <table border="1" data-bbox="464 1800 1149 1955"> <tr> <td>Size</td> <td>U8</td> <td>U16</td> <td>U16</td> <td>U16</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>P1</td> <td>I1</td> <td>D1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>U16</td> <td>U16</td> <td>U16</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[P...]</td> <td>[I...]</td> <td>[D...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and <i>PID</i> pairs can be concatenated to set different PID constants for multiple channels.</p>	Size	U8	U16	U16	U16	Field	Channel Mask 1	P1	I1	D1	Size	U8	U16	U16	U16	Field	[Channel Mask...]	[P...]	[I...]	[D...]																
Size	U8	U16	U16	U16																																	
Field	Channel Mask 1	P1	I1	D1																																	
Size	U8	U16	U16	U16																																	
Field	[Channel Mask...]	[P...]	[I...]	[D...]																																	

		<p><b>Channel Mask:</b> Bit mask selecting which channels <i>PID</i> constants will be applied to. Multiple channels can be selected by adding masks.</p> <table border="1" data-bbox="464 219 1355 293"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>P:</b> Proportional Constant. Default: 100</p> <p><b>I:</b> Integration Constant. Default: 200</p> <p><b>D:</b> Differential Constant. Default: 500</p>	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																				
Bit	7..4	3	2	1	0																													
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																													
	Reply	<table border="1" data-bbox="464 640 759 714"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																												
Size	U8																																	
Field	Return Code																																	
34 – Position Ramp		<p>Ramp to the specified position target at the specified velocity and optional acceleration rates. Control Mode must be set to <i>Position PID Control</i> using packet 16 – <i>Set Control Mode</i>.</p> <table border="1" data-bbox="464 813 1386 963"> <tr> <td>Size</td> <td>U8</td> <td>I32</td> <td>U32</td> <td>U32</td> </tr> <tr> <td>Field</td> <td>Channel Mask 1</td> <td>Position 1</td> <td>Velocity 1</td> <td>Acceleration 1</td> </tr> <tr> <td>Size</td> <td>U8</td> <td>I32</td> <td>U32</td> <td>U32</td> </tr> <tr> <td>Field</td> <td>[Channel Mask...]</td> <td>[Position...]</td> <td>[Velocity...]</td> <td>[Acceleration...]</td> </tr> </table> <p>Any number of <i>Channel Mask</i> and ramp pairs can be concatenated to initiate different ramps on multiple channels.</p> <p><b>Channel Mask:</b> Bit mask selecting on which channels to initiate position ramps. Multiple channels can be selected by adding masks.</p> <table border="1" data-bbox="464 1178 1355 1252"> <tr> <td>Bit</td> <td>7..4</td> <td>3</td> <td>2</td> <td>1</td> <td>0</td> </tr> <tr> <td>Field</td> <td>Unused</td> <td>Channel 4</td> <td>Channel 3</td> <td>Channel 2</td> <td>Channel 1</td> </tr> </table> <p><b>Position:</b> Position target.</p> <p><b>Velocity:</b> Velocity to ramp at, specified as: <i>Velocity</i> = &lt;transitions/s&gt; shl 12. The internal position target will be ramped at this rate and may be exceeded or not achieved in reality due to the acceleration limit or real-world interactions (such as re-accelerating after physically stopping the motor momentarily). The velocity limit set using packet 26 – <i>Set Velocity Limit</i> 28 – <i>Set Acceleration Limit</i> still applies. Range: 0 – 100,000 shl 12</p> <p><b>Acceleration:</b> Rate at which to accelerate towards the specified velocity, specified as: <i>Acceleration</i> = &lt;transitions/s/s&gt; shl 12. A value of zero disables acceleration limiting. The acceleration limit set using packet 28 – <i>Set Acceleration Limit</i> still applies. Range: 0 – 1,000,000 shl 12</p>	Size	U8	I32	U32	U32	Field	Channel Mask 1	Position 1	Velocity 1	Acceleration 1	Size	U8	I32	U32	U32	Field	[Channel Mask...]	[Position...]	[Velocity...]	[Acceleration...]	Bit	7..4	3	2	1	0	Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1
Size	U8	I32	U32	U32																														
Field	Channel Mask 1	Position 1	Velocity 1	Acceleration 1																														
Size	U8	I32	U32	U32																														
Field	[Channel Mask...]	[Position...]	[Velocity...]	[Acceleration...]																														
Bit	7..4	3	2	1	0																													
Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1																													
	Reply	<table border="1" data-bbox="464 1845 759 1919"> <tr> <td>Size</td> <td>U8</td> </tr> <tr> <td>Field</td> <td>Return Code</td> </tr> </table>	Size	U8	Field	Return Code																												
Size	U8																																	
Field	Return Code																																	

36 – Velocity Ramp	Ramp to the specified velocity target at the specified acceleration rate. Control Mode must be set to <i>Speed PID Control</i> using packet 16 – <i>Set Control Mode</i> .						
	Send	Size	U8	I32	U32		
Field		Channel Mask 1	Velocity 1	Acceleration 1			
		Size	U8	I32	U32		
		Field	[Channel Mask...]	[Velocity...]	[Acceleration...]		
		Any number of <i>Channel Mask</i> and ramp pairs can be concatenated to initiate different ramps on multiple channels.					
		<b>Channel Mask:</b> Bit mask selecting on which channels to initiate velocity ramps. Multiple channels can be selected by adding masks.					
		Bit	7..4	3	2	1	0
		Field	Unused	Channel 4	Channel 3	Channel 2	Channel 1
		<b>Velocity:</b> Velocity target, specified as: <i>Velocity</i> = <transitions/s>. Range: 0 - 100,000					
		<b>Acceleration:</b> Rate at which to accelerate towards the specified velocity, specified as: <i>Acceleration</i> = <transitions/s/s> shl 12. The internal velocity target will be ramped at this rate and may be exceeded or not achieved in reality due to real-world interactions (such as re-accelerating after physically stopping the motor momentarily). Set to 0 to ignore this parameter. The acceleration limit set using packet 28 – <i>Set Acceleration Limit</i> still applies. Range: 0 – 1,000,000 shl 12					
	Reply	Size	U8				
		Field	Return Code				

## Worked Examples

### Read Version Numbers

Refer to sections *Physical Connections* and *SPI Interface* and packet 1 - *Get Version* for more detail.

To query the version numbers from the QuadQuad2Drv device, send the following packet (in decimal bytes):

Meaning	STX	Packet Size	Packet ID	Checksum
Value	2	4	1	249

You should receive the following reply (in decimal bytes):

Meaning	STX	Packet Size	Packet ID	Return Code	Firmware Version Major	Firmware Version Minor	Protocol Version	Checksum
Value	2	8	1	0	0	1	0	244