Implemented?	Command Name	Description	extId	Device Type	Manufacturer	API Class	API Index	API	Mask	Device ID	Data IN Length	Data[0]	Data[1]	Data[2]	Data[3]	Data[4]	Data[5]	Data[6]	Data[7]	Data OUT Length	DataOut[0]	DataOut[1]	DataOut[2]	DataOut[3]	DataOut[4]	DataOut[5]	DataOut[6]	DataOut[7]	RTR Frame Length	Clarification Needed?	Data In	Response
Yes	Broadcast Disable	Device shall disable immediately when this meesage is recieved	0000000	deviceBroadcast	broadcast	0	0	0x00	0	ALL (ignored)																				Not implemented by NI	0 Byte Request	None
Partial	Broadcast System Halt	Upon receiving this message the motor controller will stop driving the motor and go to a neutral state. The motor can not be driven again until either a System Reset or System Resume has been received.	0000040	deviceBroadcast	broadcast	0	1	0x01	64	Individual Device																				Currently the same as Disable	0 Byte Request	None
Yes	Setpoint Set	Run the currently set control mode	2050040	motorController	REV	0	1	0x01	33882176	Individual Device	8	Target[0]	Target[1]	Target[2]	Target[3]	Aux[0]	Aux[1]	[7:3]Reserved [2] ArbFF Units pidSlot[1:0]	rsvd												Setpoint Frame	None
Yes	Duty Cycle Set	This command is used to set the output duty cycle of the motor controller in duty cycle control mode. The first parameter is a 32-bit IEEE floating point number that specifies the output duty cycle and direction. Valid range for this control mode are [-1:1].	2050080	motorController	REV	0	2	0x02	33882240	Individual Device	8	Target[0]	Target[1]	Target[2]	Target[3]	Aux[0]	Aux[1]	[7:3]Reserved [2] ArbFF Units pidSlot[1:0]	rsvd												Setpoint Frame	None
Yes	Speed Set	Sets the closed loop speed controller, unit is 32-bit IEEE floating point number representing the target speed in RPM	2050480	motorController	REV	1	2	0x12	33883264	Individual Device	8	Target[0]	Target[1]	Target[2]	Target[3]	Aux[0]	Aux[1]	[7:3]Reserved [2] ArbFF Units pidSlot[1:0]	rsvd												Setpoint Frame	None
Yes	Smart Velocity Set	Sets the closed loop speed controller, unit is 32-bit IEEE floating point number representing the target speed in RPM. Honors the max acceleration and max velocity from the Smart Motion parameters	20504C0	motorController	REV	1	3	0x13	33883328	Individual Device	8	Target[0]	Target[1]	Target[2]	Target[3]	Aux[0]	Aux[1]	[7:3]Reserved [2] ArbFF Units pidSlot[1:0]	rsvd													
Yes	Position Set	Sets the closed loop speed controller, unit is 32-bit IEEE floating point number representing the target position in rotations	2050C80	motorController	REV	3	2	0x32	33885312	Individual Device	8	Target[0]	Target[1]	Target[2]	Target[3]	Aux[0]	Aux[1]	[7:3]Reserved [2] ArbFF Units pidSlot[1:0]	rsvd												Setpoint Frame	None
Yes	Voltage Set	Sets the closed loop speed controller, unit is 32-bit IEEE floating point number representing the target voltage in volts.	2051080	motorController	REV	4	2	0x42	33886336	Individual Device	8	Target[0]	Target[1]	Target[2]	Target[3]	Aux[0]	Aux[1]	[7:3]Reserved [2] ArbFF Units pidSlot[1:0]	rsvd												Setpoint Frame	None
Yes	Current Set	Sets the closed loop speed controller, unit is 32-bit IEEE floating point number representing the target current in Amps.	20510C0	motorController	REV	4	3	0x43	33886400	Individual Device	8	Target[0]	Target[1]	Target[2]	Target[3]	Aux[0]	Aux[1]	[7:3]Reserved [2] ArbFF Units pidSlot[1:0]	rsvd													
Yes	Smart Motion Set	Sets the closed loop smart motion controller, unit is 32-bit IEEE floating point representing the target position in rotations	2051480	motorController	REV	5	2	0x52	33887360	Individual Device	8	Target[0]	Target[1]	Target[2]	Target[3]	Aux[0]	Aux[1]	[7:3]Reserved [2] ArbFF Units pidSlot[1:0]	rsvd													
Partial	Periodic Status 0	Set status frame period - default 10ms	2051800	motorController	REV	6	0	0x60	33888256	Individual Device	2	**Frame Period ms[0]**	**Frame Period ms[1]**							8	Applied Output LSB	Applied Output MSB	Faults LSB	Faults MSB	Sticky Faults LSB	Sticky Faults MSB	Rsvd	Invert/Brake Settings, Is Follower			No response to calling this frame direction. Status 0 frame out is periodic	Status In Frame (Set frame period)	None
Partial	Periodic Status 1	Set status frame period - default 20ms All data is 16-bit half precision numbers	2051840	motorController	REV	6	1	0x61	33888320	Individual Device	2	**Frame Period ms[0]**	**Frame Period ms[1]**							8	Motor Velocity LSB	Motor Velocity MID_L	Motor Velocity MID_H	Motor Velocity MSB	Motor Temperature	Motor Voltage LSB	Motor Current LSB 4 bits Motor Voltage MSB 4 bits	Motor Current MSB				Status In Frame (Set frame period)	None
Partial	Periodic Status 2	Set user frame period - default 0 (disabled) Each user frame can output four 16-bit parameters	2051880	motorController	REV	6	2	0x62	33888384	Individual Device	2	**Frame Period ms[0]**	**Frame Period ms[1]**							8	Motor Position LSB	Motor Position MID_L	Motor Position MID_H	Motor Position MSB	RSVD	RSVD	RSVD	RSVD					
1.4.0 and above	Periodic Status 3	Set user frame period - default 0 (disabled) Each user frame can output four 16-bit parameters	20518C0	motorController	REV	6	3	0x63	33888448	Individual Device	2	**Frame Period ms[0]**	**Frame Period ms[1]**							8	adcVoltage 2q8 [7:0]	adcVoltage 2q8 [8:9] analogVelocity 15q7[0:7]	analogVelocity 15q7[8:15]	analogVelocity 15q7[16:22]	analogPos IEEE Float LSB	analogPos IEEE Float MID LOW	analogPos IEEE Float MID HIGH	analogPos IEEE Float MSB					
1.5.0 and above	Periodic Status 4	Set user frame period - default 0 (disabled) Each user frame can output four 16-bit parameters	2051900	motorController	REV	6	4	0x64	33888512	Individual Device	2	**Frame Period ms[0]**	**Frame Period ms[1]**							8	alt encoder velocity IEEE Float LSB	alt encoder velocity IEEE Float MID LOW	alt encoder velocity IEEE Float MID HIGH	alt encoder velocity IEEE Float MSB	alt encoder pos IEEE Float LSB	alt encoder pos IEEE Float MID LOW	alt encoder pos IEEE Float MID HIGH	alt encoder pos IEEE Float MSB					
Yes	DRV Status	The first four bytes contain the status directly from the SPI of the DRV8S2x (STAT0 and STAT1). See datasheet for more details http://www.ti.com/lit/ds/symlink/drv8320.pdf The second 4 bytes contain the faults and sticky faults as a way to poll instead of relying on the periodic messages.	2051A80	motorController	REV	6	10	0x6A	33888896	Individual Device	0									8	STAT0 LSB	STAT0 MSB	STAT1 LSB	STAT1 MSB	**Faults LSB**	**Faults MSB**	**Sticky Faults LSB**	**Sticky Faults MSB**			Status from SPI of DRV and faults	0 Byte Request	DRV Status Frame
Yes	Clear Faults	Clear sticky faults	2051B80	motorController	REV	6	14	0x6E	33889152	Individual Device	0																				0 Byte Request	None
Yes	Config Burn Flash	Burns flash updating only parameters that changed. Can only be done when device is not enabled (for now?).	2051C80	motorController	REV	7	2	0x72	33889408	Individual Device	2	0xA3	0x3A																		0 Byte Request	None
Yes	Set Follower Mode		2051CC0	motorController	REV	7	3	0x73	33889472	Individual Device	8	FollowerID[0]	FollowerID[1]	FollowerID[2]	FollowerID[3]	FollowerCfg[0]	FollowerCfg[1]	FollowerCfg[2]	FollowerCfg[3]	8	FollowerID[0]	FollowerID[1]	FollowerID[2]	FollowerID[3]	FollowerCfg[0]	FollowerCfg[1]	FollowerCfg[2]	FollowerCfg[3]		Is Follower' bit of status frame should indicate success	8 Byte Request	None
Yes	Config Factory Defaults	Reset most parameters to their factory defaults - This API does nt overwrite certain parameters outlined in the parameter table. Send a boolean 'True' to also burn the parameter table.	2051D00	motorController	REV	7	4	0x74	33889536	Individual Device	5	Burn Table?	0	0	0	Parameter Type Bool																
Yes	Config Factory Reset	Reset all parameters to their factory defaults, this include CAN ID and others. Send a boolean 'True' to also burn the parameter table.	2051D40	motorController	REV	7	5	0x75	33889600	Individual Device	5	Burn Table?	0	0	0	Parameter Type Bool																
1.5.0 and above	Identify	Present a visual blink code on the LEDs to identify a particular device	2051D80	motorController	REV	7	6	0x76	33889664	Individual Device	4	Unique ID (If CAN ID = 0)[0]	Unique ID (If CAN ID = 0)[1]	Unique ID (If CAN ID = 0)[2]	Unique ID (If CAN ID = 0)[3]																		
No	Nack	General Non-acknowledgement	2052000	motorController	REV	8	0	0x80	33890304	Individual Device																				Not implemented	Not implemented	Not implemented
Yes	Ack	General Acknowledge (used by USB)	2052040	motorController	REV	8	1	0x81	33890368	Individual Device										0												
N/A	Broadcast (not a command)		2052400	motorController	REV	9	0	0x90	33891328	Individual Device	0									0												
Yes - INTERNAL ONLY	Heartbeat	Heartbeat command for all REV motor controllers	2052480	motorController	REV	9	2	0x92	33891456	Any REV Motor Controller	8	Enabled[0]	Enabled[1]	Enabled[2]	Enabled[3]	Enabled[4]	Enabled[5]	Enabled[6]	Enabled[7]												Heartbeat command	None
Yes	Sync	Synchronize all REV motor controllers	20524C0	motorController	REV	9	3	0x93	33891520	Any REV Motor Controller	0																				0 Byte Request	None
1.5.0 and above	IDQuery	Cause all REV motor controllers whos IDs are set to 0 to all respond with a hashed version of their serial numbers (96-bit unique number hash to 48 bits) after a random number of ms. Arbitration/auto retry means all messages will get through on the bus. Controller now has a list of all hashed IDs on the bus and can address by this 48-bit id - collision is possible but unlikely	2052500	motorController	REV	9	4	0x94	33891584	Any REV Motor Controller	0									4	ID[0]	ID[1]	ID[2]	ID[3]						0 Byte Request	ID Result
1.5.0 and above	IDAssign	Use 32-bit hashed unique ID to assign the CAN ID of the controller	2052540	motorController	REV	9	5	0x95	33891648	Any REV Motor Controller	5	ID[0]	ID[1]	ID[2]	ID[3]	CANID[0]															0 Byte Request	None
Yes Updated for v1.5.0 Updated for v1.6.0	Firmware Version	This command is sent to request the current firmware version for the motor controller. This command uniquely addresses a device and only the addressed device will respond to this message. The motor controller will send back four bytes of data that indicate the firmware version of the motor controller and one byte indicating if this is a debug or release build.	2052600	motorController	REV	9	8	0x98	33891840	Individual Device	0									6	Firmware Version	Firmware Version	Firmware Version	Firmware Version	is debug?	**HW Rev (ASCII Char)**	**Firmware Hash [0]**	**Firmware Hash [1]**	6	This is a duplicate of the boardcast frame in REV address space	0 Byte Request	Firmware Frame
No	REV Enumerate	This command causes the motor controller to send out a response to indicate that device is present on the CAN network. In order to prevent all devices from responding at once, the motor controllers will wait for (device number) + 1 ms after the enumerate command before responding. Once enumeration has been started, the CAN device that requested the enumeration sequence should wait at least 80ms before generating any other CAN traffic to avoid affecting the enumeration sequence. After the enumeration sequence is complete, normal CAN activity should resume allowing the motor controllers to keep their CAN links active. The motor controller will also send out an enumeration message with its ID when it is first started. This can be used by the CAN controller to detect when new motor controllers become available, and to detect when existing motor controllers are restarted because of an intermittent power failure.	2052640	motorController	REV	9	9	0x99	33891904	Individual Device	0									0												
1.3.0 and above	roboRIO Lock	Command from roboRIO which 'locks' the device into roboRIO mode. If this frame is recieved and is valid the device will only accept valid heartbeat frames from the roboRIO. (Latches the device into 'competition FW mode' or similar). Lock type is valid for what is allowed to be set out by the USB bus.	20528C0	motorController	REV	9	11	0x0B	33892032	Individual Device	8	API[0]	API[1]	API[2]	API[3]	LockType (Default = 0 = Lock commands AND heartbeat) 1 = Lock out only heartbeat	0	0	0													
Yes	Telemetry Update Mechanical Position Enoder Pos	Manually set the telemetry data of the controller	2052800	motorController	REV	10	0	0xA0	33892352	Individual Device	8	MechPos[0]	MechPos[1]	MechPos[2]	MechPos[3]																	
Yes	Telemetry Update I Accum	Manually set the telemetry data of the controller	2052880	motorController	REV	10	2	0xA2	33892480	Individual Device	8	IAccum[0]	IAccum[1]	IAccum[2]	IAccum[3]																	
1.4.0 and above	Telemetry Update Mechanical Position Analog	Manually set the telemetry data of the controller	20528C0	motorController	REV	10	3	0xA3	33892544	Individual Device	8	MechPos[0]	MechPos[1]	MechPos[2]	MechPos[3]																	
1.4.0 and above	Telemetry Update Mechanical Position Alt Encoder		2052900	motorController	REV	10	4	0xA4	33892608	Individual Device																						
N/A	Non-roboRIO Broadcast (not a command)	Not a command, this group is for non-roboRIO useage broadcast commands	2052C00	motorController	REV	11	0	0xB0	33893376	Individual Device	8	Enabled[0]	Enabled[1]	Enabled[2]	Enabled[3]	Enabled[4]	Enabled[5]	Enabled[6]	Enabled[7]													

Implemented?	Command Name	Description	extId	Device Type	Manufacturer	API Class	API Index	API	Mask	Device ID	Data IN Length	Data[0]	Data[1]	Data[2]	Data[3]	Data[4]	Data[5]	Data[6]	Data[7]	Data OUT Length	DataOut[0]	DataOut[1]	DataOut[2]	DataOut[3]	DataOut[4]	DataOut[5]	DataOut[6]	DataOut[7]	RTR Frame Length	Clarification Needed?	Data In	Response
1.3.0 and above	non-RIO Lock	Command from other processor which locks out USB from sending command or heartbeat frames. This is ignored if the SOURCE of the command is our own USB. This is useful for setting a device on the bus as the 'master'. For example a Raspberry Pi can lock down the bus as the owner of the device to prevent other devices from commanding SPARK MAX's.	2052C40	motorController	REV	11	1	0xB1	33893440	Individual Device	8	API[0]	API[1]	API[2]	API[3]	LockType (Default = 0 = Lock commands AND heartbeat) 1 = Lock out only heartbeat	0	0	0													
1.3.0 and above	non-RIO Heartbeat	Heartbeat command for all REV motor controllers. This is the same as the heartbeat but does not activate the controller if a lock packet has been recieved. This command waits for an additional one second after boot to check for rio lock.	2052C80	motorController	REV	11	2	0xB2	33893504	Individual Device	8	Enabled[0]	Enabled[1]	Enabled[2]	Enabled[3]	Enabled[4]	Enabled[5]	Enabled[6]	Enabled[7]													
1.5.0 and above	USB Only Identify		2052CC0	motorController	REV	11	3	0xB3	33893568	Individual Device	0									0												
Yes	Parameter Access	Set parameter using the CAN ID fields instead of a selection in the packet	205C000	motorController	REV	48	0	0x300	33931264	Individual Device	5	Param[0]	Param[1]	Param[2]	Param[3]	Parameter Type				6	Param[0]	Param[1]	Param[2]	Param[3]	Parameter Type	Parameter Response [0 = response OK]			6	This API is o/d with the parameter ID. Send a 0 data length message and/or send the remote bit to get the value, send data to set it.		