Source	Dashboard		
Description	Vehicle state		
Bus	PCAN		
CAN ID	0x766		
Length	7 bytes		
0	HV requested	0 = not requested, 1 = requested	
1	Throttle 1 Level	Throttle1 %	
2	Throttle 2 Level	Throttle2 %	
3	Brake Level	Brake %	
4	Vehicle State	LV	0000 0000 = 0x00
		Precharge	0000 0001 = 0x01
		HV	0000 0010 = 0x02
		Drive	0000 0011 = 0x03
		Startup	0000 0101 = 0x05
		FAULT: Drive request from LV	1000 0001 = 0x81
		FAULT: Precharge timeout	1000 0010 = 0x82
		FAULT: Brake not pressed	1000 0011 = 0x83
		FAULT: HV disabled while driving	1000 0100 = 0x84
		FAULT: Sensor discrepancy	1000 0101 = 0x85
		FAULT: BSPD tripped	1000 0110 = 0x86
		FAULT: Shutdown Circuit Open	1000 0111 = 0x87
		FAULT: Uncalibrated	1000 1000 = 0x88
		FAULT: Hard BSPD	1000 1001 = 0x89
		FAULT: MC Fault	1000 1010 = 0x8A
5	VCU ticks upper		
6	VCU ticks lower	uint16_t, in millseconds, for SavvyCAN's broke timestamp	

Source	Dashboard		
Description	Torque Request		
Bus	PCAN		
CAN ID	0x0C0		
Length	8 bytes		
0	Torque lower		
1	Torque upper	Torque in Nm times 10, signed 16 bit	
2	Speed lower	Angular velocity in RPM	
3	Speed upper		
4	Direction command	0 = Reverse, 1 = Forward	
5	Bit 0: Inverter Enable	0 = Disable, 1 = Enable. Equivalent of hv_enable()?	

	Bit 1: Discharge Enable	Always have discharge enabled?	
	Bit 2: Speed mode enable	Always disable	
6	Torque limit lower	Torque in Nm times 10	
7	Torque limit upper	0 = Use default	

Source	Dash		
Description	Random shit for testing		
CAN ID	0x500		
Length	6 bytes		
0	front strain gauge (Upper 8)	uint16_t, raw ADC	
1	front strain gauge (lower 8 8)		
2	Front wheel speed (Upper 8)	uint16_t, rpm	
3	Front wheel speed (Lower 8)		
4	TC Torque Request (Upper 8)	uint16_t, Nm * 10	
5	TC Torque Request (Lower 8)		

Source	Dashboard		
Description	Dashboard Inputs		
Bus	PCAN		
CAN ID	0x501		
Length	5 bytes		
0	Knob 1 (Upper 8)	uint16_t, raw ADC	
1	Knob 1 (Lower 8)		
2	Knob 2 (Upper 8)	uint16_t, raw ADC	
3	Knob 2 (Lower 8)		
4	Dashboard Mode	uint8_t (0x00 = DRIVE MODE, 0x01 = DEBUG MODE)	

Source	PEI		
Description	PEI Status		
Bus	PCAN		
CAN ID	0x387		
Length	5 bytes		
0	Shutdown circuit flags	00| IMD_OK | BMS_OK | Shutdown Final | AIR_NEG | AIR_POS | Precharge	
1	Current ADC Reading (Upper 8 bits)	Subtract current reference from this after converting both to amps	
2	Current ADC Reading (Lower 8 bits)		
3	Current Reference ADC Reading (Upper 8 bits)		
4	Current Reference ADC Reading (Lower 8 bits)		

Source	PEI		
Description	BMS Status		
Bus	PCAN		
CAN ID	0x380		
Length	5 bytes		
0	BMS Status	Normal/No Error	0000 0000 = 0x00
		Chargemode	0000 0001 = 0x01
		FAULT: Pack temp over	0000 0010 = 0x02
		FAULT: Pack temp under	0000 0100 = 0x04
		FAULT: Cell volt over	0000 1000 = 0x08
		FAULT: Cell volt under	0001 0000 = 0x10
		FAULT: Open Wire - Off-board disconnect between cell and BMS IC	0010 0000 = 0x20
		FAULT: Mismatch - Discrepancy between main ADC and redundant ADC cell voltage readings (likely due to on-board disconnect between cell and BMS IC)	0100 0000 = 0x40
		FAULT: SPI fault	1000 0000 = 0x80
1	SPI error flags (Upper 8 bits)	Error flags (one for each BMS IC, of which there are 10)	
2	SPI error flags (Lower 8 bits)		
3	Max faulting BMS IC address	Number from 0 to 9	
4	Max faults	Number of faults on max faulting BMS IC address	
5	Communication break ID	Address of IC where communication break occurred (signed 8 bit, -1 when there's no communication break	

Source	PEI		
Description	Diagnostic BMS Data		
Bus	PCAN		
CAN ID	0x381		
Length	4 bytes		
0	HI temp	Maximum temp in C, unsigned 8 bit	
1	SOC	State of charge (%), unsigned 8 bit	
2	Pack voltage (Upper 8 bits)	Pack voltage in raw signed 16 bit form	
3	Pack voltage (Lower 8 bits)		

Source	Raspberry Pi		
Description	Raspberry Pi AD HAT Data (logged to Raspberry Pi SD card only, not an actual CAN message)		
ID	0x382		
	AD HAT Input	CSV column	
	Input 0	D0	
	Input 1	D1	
	Input 2	D2	

	Input 3	D3	
	Input 4	D4	
	Input 5	D5	
Source	Motor Controller		

CM200 MC CAN docs
https://app.box.com/s/vf9259qlaadhzxqiqrt5cco8xpsn84hk/file/27334613044
Newer Software Manual that includes CAN docs
https://www.cascadiamotion.com/uploads/5/1/3/0/51309945/0a-0163-02_sw_user_manual.pdf

3 types of CAN messages:
- Broadcast messages: from MC. continously send data about itself
- Parameter messages: from another node. read/write parameters
- Command message(ID: 0x0C0): from another node. commands the MC. most notably torque request
    - Sent from Dashboard in our system (see above)

ALL MESSAGES ARE LITTLE ENDIAN