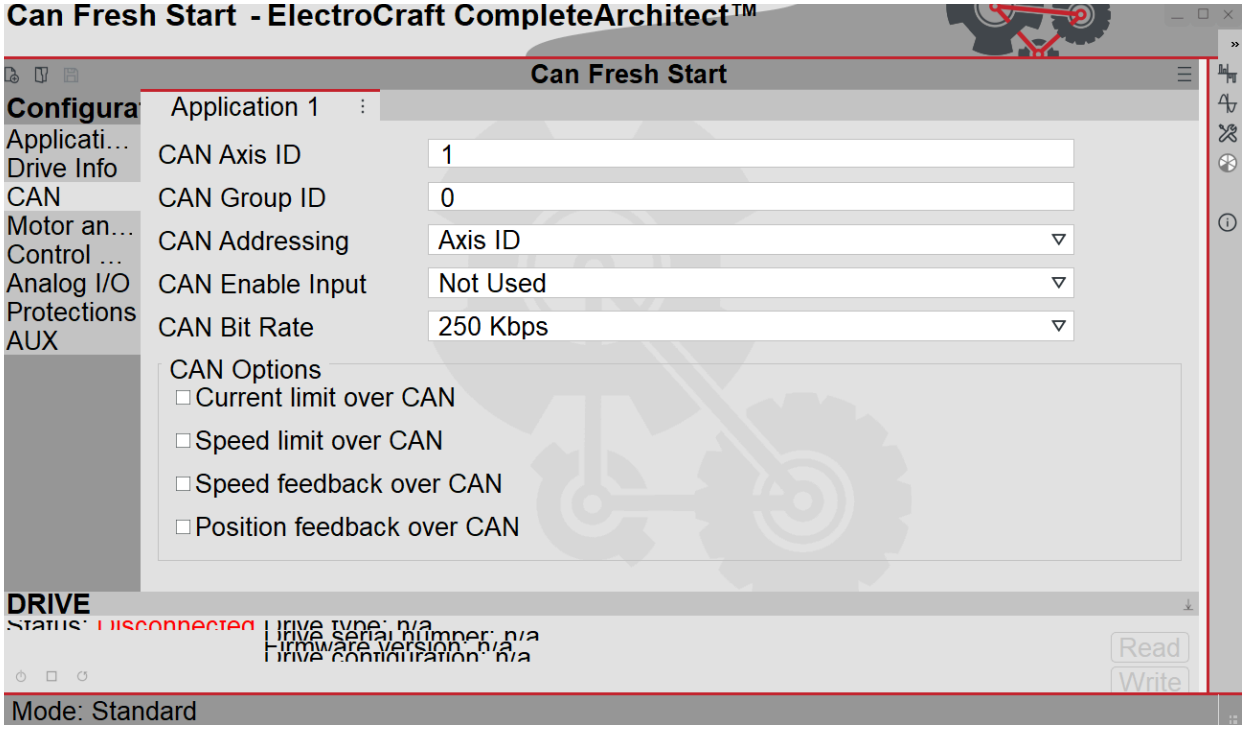
Electrocraft CAN Bus Report



The main point to explain here is the *CAN Enable Input*. Four options in that drop-down menu: Not Used, Enable CAN, Enable Drive, and Enable CAN and Drive. Setting this up, the <u>CAN Enable input must be set to Not Used.</u> The other options are referring to the EN IN + and EN IN - in this diagram:

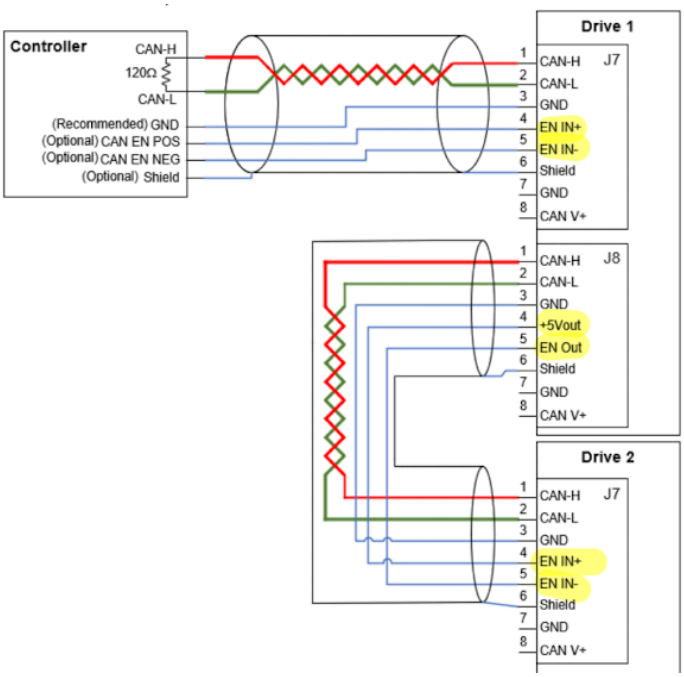
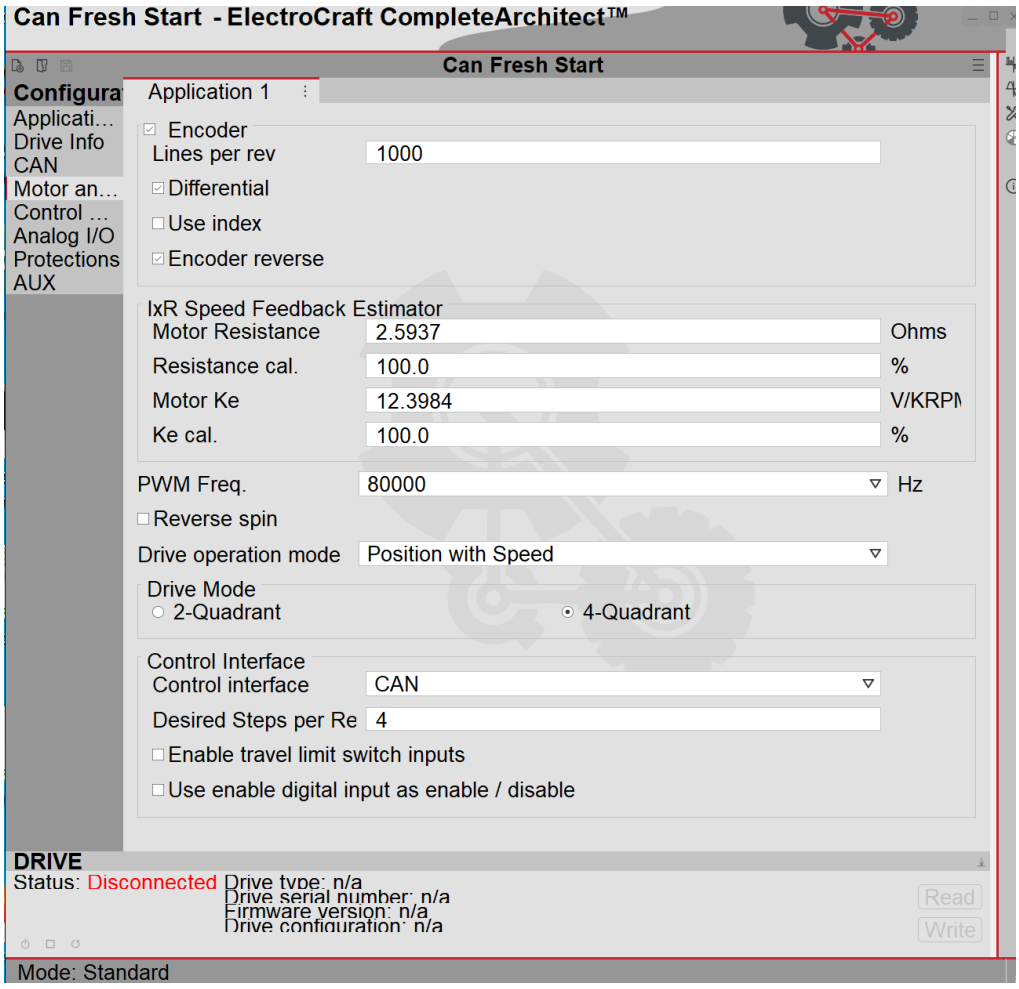


Figure 1: CAN Ethernet port on the side of the drive.

Going more in-depth, depending on the option selected for *CAN Enable Input* EN IN + and EN IN - will enable the CAN bus, the drive, or both if 3.3V (Thomas mentioned 3.3V, documentation days 5V+) goes through it. The reason for this design choice is because the drives are meant to be “daisy chained” (look at the diagram and how drive 1 connects to drive 2. It doesn't have to stop at 2 it can be going to three, four, etc). In other words, put 3.3V into EN IN + and if wired properly it should enable all the drives connected in the CAN network. Make sure the wiring is right. CAN High, CAN Low, and ground must be connected properly for the acknowledgment (handshake between interface and CAN bus confirming the message was properly received or not) to happen.

CAN Options has multiple checkboxes that are left unchecked because in the manual it refers to them as feedback, not commands. It might be for a CAN-based encoder of some sort, but for this purpose, it should not be relevant. Moving on to the next page.



Drive operation mode alters the behavior of these commands:

3.7 Move Command

The MOVE command can be used to instruct the drive to move (or stop) the motor. The nature of the move (or stop) depends on the command code provided. See Table 1.7 for more details.

Description	Type	Service Bit	Req/!Resp Bit	Service ID (Hex)
MOVE Command	Tx	1	1	0x41
MOVE Command Status Reply	Rx	1	0	0x41

Description	DLC	Data1	Data2	Data3	Data4	Data5	Data6	Data7	Data8
MOVE Command	0x05	CMD	Byte0	Byte1	Byte2	Byte3	-	-	-
MOVE Command Status Reply	0x02	Status	CMD	-	-	-	-	-	-

Name	CMD	Size	Description
Move Absolute Immediate	0x00	int32	Immediate step move to new position without trajectory control.
Move Absolute	0x01	int32	Move to specified position using trapezoidal trajectory.
Move Relative Immediate	0x02	int32	Immediate move of a specified number of encoder counts from current position.
Move Relative	0x03	int32	Move a specified number of encoder counts from current position using trapezoidal trajectory.
Current Move	0x10	int16	Set drive current output to specified value.
Speed Move	0x20	int32	Set drive speed setpoint to specified value.
Abort Torque Off	0x80	-	Abort current move by turning off torque to motor.
Abort Decel. to Stop	0x81	-	Abort current move. Decelerate to zero using current trajectory parameters. Hold position when stopped.

Figure 2: CAN Reference Manual move commands

How the options (Current, Speed, Position, Position with Speed) affect the commands is to be determined, but *Position and Speed* seem to work the best for *Move Relative Immediate* as it seems to use Saad's control loop parameters. (I probably butchered control loop terminology… oops)

Control Interface needs to be set to CAN. "Use enable digital input as enable/disable" and "Enable Travel limit switch

inputs" needs to be unchecked. I am not sure whether "Enable Travel limit switch inputs" actually needs to be unchecked but I do know that if you want the motor to move through CAN you have to enable it through CAN. Enabling the drive through the I/O connector and then sending a message through CAN, the drive will acknowledge the data frame (CAN message) but the drive will not move the motor. Important note manual does warn the I/O connector might interfere with CAN so going through the hassle to unplug the controller from that very annoying connector might help.

Special thanks to Thomas (1st Gen Co-bot team) for providing information and help.