Direct Control

Table of Contents

Export to PDF

Introduction

All the protocols can use the Direct Control method. Using Direct Control allows the user to directly control the motors position, torque and speed. All values can be updated dynamically which allows for very flexible operation. Not only does Direct Control allow you to control the motor but the read registers give a full array of motor information for program use and diagnostics.The CML control method has a special provision to run Direct Control by writing to specific CML registers. The documentation below describes how to use Direct Control. For details on each of the protocols please see their specific sections

- Modbus TCP
- Direct Control Port
- EtherNet/IP - EtherNet/IP includes AOIs that setup all required mappings.
- CML - The CML registers used are detailed below in the Direct Control documentation

Though all the protocols utilize the same set of variables they can work concurrently. If multiple protocols are to be used concurrently then the user should be aware of how the write operations function:

- EtherNet/IP - Cyclical (1ms)
- Modbus-TCP - On write
- TCP/UDP - On write

As can be seen above, if EtherNet/IP is used it will overwrite any of the other protocols in a write operation. The other protocols can still be used for read operations and diagnostics.

Read Registers

Register	Description
CPUTime	CPU clock time in 50us counts
ActualPosition	The actual position of the motor in encoder counts
ActualTargetPosition	The actual instantaneous position the motor is currently moving to.
MotorStatus	The motors status (error, homing, in position, etc)

ActualRatedCurrent	Percentage (0.1%) of rated current. 1000 = rated, 1100 = peak.
ActualOverloadTorque	Percentage (0.1%) of overload torque.
AnalogIN	10 bit analog input value (0-1023)
DigitalIO	Digital IO status. • B3-B0 = IN4-IN1 • B5-B4 = OUT2 - OUT1
Temperature	Drive temperature in ºC
DCVoltage	24V DC bus voltage in 0.1V
DigitalOUT	Digital OUT status. B0-B1 = OUT1-OUT2
ModeOfOperationDisplay	Indicates the mode of operation currently set
ActualSpeed	Actual speed of the motor in encoder counts/s

Write Registers

Register	Description
TargetPosition	Final target position
TargetSpeed	Maximum speed
TargetTorque	Maximum torque
TargetAcceleration	Acceleration (used when accelerating to target speed)
TargetDeceleration	Deceleration (used when stopping)
Controlword	Control the motor operation
ModeOfOperation	Set the required mode of operation
DigitalOUT	Set the 2 digital outputs. Requires K34=44.

Setting Mode Of Operation

There are a number of modes of operation the motor can be set to.

1. CML mode - this is the standard and default mode. If no mode is selected, this is the mode the motor Cool Muscle operates in.
2. Profile mode - run position or speed profiles.
3. Torque mode - run the motor by adjusting the torque and maximum speed values.

The control registers in the mode selected use either standard CML registers or registers set directly such as through Modbus-TCP. The mode of operation selected indicates where the control registers will be received

The following table lists the mode of operation and the associated registers

Mode of Operation	Value	Description
CML Mode	0	Standard CML mode. No control registers associated
Profile Mode	2	Profile mode using P1, S1, A1, A2, R1 and N1
	3	Profile mode using • Modbus control registers (see registers starting 41029). • EtherNet/IP IO messaging • TCP/UDP port 10002
Dynamic Position Mode	4	Dynamic position mode using P1, A1, R1 and N1

	5	Dynamic position mode using • Modbus control registers (see registers starting 41029). • EtherNet/IP IO messaging • TCP/UDP port 10002
Torque Mode	10	Torque mode using N1, S1 and R1
	11	Torque mode using • Modbus control registers (see registers starting 41029). • EtherNet/IP IO messaging • TCP/UDP port 10002

- Bit0 of the mode of operation selects the register source. I.e. P1 to N1 or the direct control registers.

Set the mode of operation by the following

1. CML
    a. Send "_mop=x" on the CML port when x is the value of the mode of operation
    b. Assign a variable to "mop" and write the variable in a logic bank. E.g. V0="mop".
2. Direct
    a. Set modbus register 41041 to the value of the required mode of operation.
    b. Set Byte[16] in EtherNet/IP
    c. Set Byte[16] in TCP/UDP Port 10002

Registers

The following registers are used in the different Direct Control modes of operation.

Controlword

The Controlword is used to control the motor. The usage does depend on the mode that is selected. See the mode for bit usage. B4-B7 are common to all modes.

Register Name	Profile Mode = 2	Profile Mode = 3			Unit	Description
	CML Registers	Modbus Registers	TCP/UPD Bytes	Ethernet/IP O2T Bytes		
Controlword	R1	41039	14-15	14-15	N/A	See below

Bit Descriptions:

BIT	Name	Value	Description
B0	Operation mode specific		
B1	Operation mode specific		
B2	Operation mode specific		
B3	Operation mode specific		
B4	Disable	0 1 1 0	Enable the motor Disable the motor Reset any error
B5	Reset Error	0 1	Reset any error
B6	Set to Zero	0 1	Set the current position to 0
B7	Home	0 1	Start a home search(1)

(1) Set the home search parameters with K parameters K42-K48

Statusword

The Statusword returns information on the status of the motor.

Register Name	Profile Mode = 2	Profile Mode = 3			Unit	Description
	CML Registers	Modbus Registers	TCP/UPD Bytes	Ethernet/IP O2T Bytes		
Statusword	Query ?99	40009	16-17	16-17	N/A	See below

Bit Descriptions:

BIT	Name
B0	Position overflow alarm
B1	Overspeed alarm
B2	Overload alarm
B3	Inposition
B4	Disable
B5	Pushmode limit
B6	Communication error alarm
B7	Over temperature alarm
B8	Pushmode Timeout warning
B9	E-Stop active
B10	N/A
B11	N/A
B12	N/A
B13	N/A
B14	New start required
B15	Home achieved

TargetPosition

The TargetPosition sets a new target position for the motor.

Register Name	Profile Mode = 2	Profile Mode = 3			Unit	Range
	CML Registers	Modbus Registers	TCP /UPD Bytes	Ethernet /IP O2T Bytes		
TargetPosition	P1	41029	0-3	0-3	pulses (counts)	$-2^{\wedge 31}$ to $+2^{\wedge 31}$

The motor has a maximum resolution of 50,000 counts per revolution. Regardless of the resolution defined in K37 the motor internally will convert the target position to a position based on 50,000. As such the maximum range depends on the motor resolution. For example if K37=3 and the resolution is 1000 pulses/revolution, the maximum target position is $2^{\wedge 31} \div (50{,}000/1{,}000) = 2^{\wedge 31} \div 50 = 42{,}949{,}672$.

TargetSpeed

The TargetSpeed sets a new maximum speed.

Register Name	Profile Mode = 2	Profile Mode = 3			Unit	Range
	CML Registers	Modbus Registers	TCP/UPD Bytes	Ethernet/IP O2T Bytes		

TargetSpeed	S1	41031	4-7	4-7	pulses/second 10 pulses/second 100 pulses/second	$-2^{\wedge 31}$ to $+2^{\wedge 31}$

The unit depends on the motor resolution (K37) value selected. See K37 for more details.

TargetTorque

The TargetTorque sets the maximum torque used by the motor.

Register Name	Profile Mode = 2	Profile Mode = 3			Unit	Range
	CML Registers	Modbus Registers	TCP/UPD Bytes	Ethernet/IP O2T Bytes		
TargetTorque	N1	41033	8-9	8-9	0.1% rated torque	0-1100 (±1100 in Torque mode)

The torque is set in a percentage of rated torque. The Cool Muscle will only use the torque required so typically this value should be set to 1100 unless the application requires that the torque is limited.

TargetAcceleration

The TargetAcceleration sets the acceleration used by the motion profile.

Register Name	Profile Mode = 2	Profile Mode = 3			Unit	Range
	CML Registers	Modbus Registers	TCP /UPD Bytes	Ethernet /IP IO Bytes		
TargetAcceleration	A1	41035	10-11	10-11	Kpulses/second^2	± 32767

If the TargetAcceleration=0 the profile shall be undefined and may result in an unexpected motion profile.

TargetDeceleration

Register Name	Profile Mode = 2	Profile Mode = 3			Unit	Range
	CML Registers	Modbus Registers	TCP /UPD Bytes	Ethernet /IP O2T Bytes		
TargetDeceleration	A2	41037	12-13	12-13	Kpulses/second^2	± 32767

If the TargetDeceleration=0 the TargetAcceleration shall be used. This allows the user to only change the acceleration if a trapezoid type profile is required.

Modes of Operation

CML Mode

CML mode is the standard mode that the Cool Muscle motor typically operates in. There are no specific control registers. See the CM1 documentation for CML mode usage.

Profile Mode (Position and Speed)

Profile mode would be a typical mode used in point-to-point motion and/or continuous velocity motion. The following registers are used depending on the profile mode selected.

Register Name	Profile Mode = 2	Profile Mode = 3			Description
	CML Registers	Modbus Registers(1)	TCP/UPD Bytes(2)	Ethernet/IP O2T Bytes(3)	

TargetPosition	P1	41029	0-3	0-3	The target position the motor will move to. • Used as an absolute position • Used as an incremental position in a relative control(1) move • Not used in speed control(1) (1) See the Controlword for switching to speed or relative control.
TargetSpeed	S1	41031	4-7	4-7	The maximum speed of the profile trajectory. • the sign of S1 is ignored in a position control move. • +ve S1 in speed control will run the motor in a +ve direction (typically CW) • -ve S1 in speed control will run the motor in a -ve direction (typically CCW)
TargetTorque	N1	41033	8-9	8-9	The maximum torque the motor will use.
TargetAcceleration	A1	41035	10-11	10-11	Acceleration is used when the profile is accelerating to the target speed. This is regardless of whether the target speed will be reached or not
TargetDeceleration	A2	41037	12-13	12-13	Deceleration is used when the profile is decelerating to the target position.
Controlword	R1	41039	14-15	14-15	Bits in the Controlword dictate profile execution. A more detailed description can be found below.
ModeOfOperation	Set with "_mop"	41041	16	16	Sets the mode of operation see Setting Mode of Operation
DigitalOUT	N/A	41043	17	17	Binary value to set the digital output (Register K34=44 to enable this function)

(1) See Modbus TCP for additional details
(2) See TCP/UDP Port 10002 for additional details
(3) See EtherNet/IP for additional details. EtherNet/IP includes AOIs that setup all required mappings.

Controlword Usage

Profile mode uses some specific bits in the Controlword

BIT	Name	Value	Description
B0	Start/New Set Point	0	• Do not execute any changes in profile registers • Do not start a new profile • Any profile currently in motion will be completed

		0 1	• Start a profile move defined by the profile registers
			• 0 1 transition is required to start a profile after
			  ◦ an error has been cleared
			  ◦ a home has been completed
			  ◦ a stop command has been given from CML or an input.
			• In relative mode transition is required to start the next incremental move.
			  ◦ The move is incremented from the current target position not the actual position.

		1	Execute any changes in any of the profile registers immediately.
B1	Halt	0	Do nothing
		1	• Halt the current profile being executed • Prohibit execution of a profile if no profile is being run
B2	Relative mode	0	TargetPosition is an absolute position
		1	TargetPosition is a relative incremental position
B3	Speed control	0	Position control mode
		1	Speed control mode
B7-B4	See Controlword		

Profile Mode Examples

The following example uses the standard registers to run an absolute move. This could be run from a Control Room CML script.

Absolute Move Example

```
R1=0                //clear the controlword before setting the mode
_mop=2                //set the mode of operation to 2. Profile mode
P1=10000        //set target position
S1=100                //set target speed
A1=10                //set target acceleration
A2=10                //set target deceleration
N1=1100                //set the target torque to peak
R1=1                //start the profile move

//While R1=1 any change in the profile registers will execute an immediate change
```

The following example uses the standard registers to run a speed move. This could be run from a Control Room CML script.

Speed Move Example

```
R1=0                //clear the controlword before setting the mode
_mop=2                //set the mode of operation to 2. Profile mode
S1=100                //set target speed
A1=10                //set target acceleration
A2=10                //set target deceleration
N1=1100                //set the target torque to peak
R1=9                //start the profile move with the speed mode bit set.

//While R1=9 any change in the profile registers will execute an immediate change
```

Dynamic Position Mode

This mode can be used to dynamically stream a target position only and have the motor track it. There is a filter that filters the responsiveness of the position change as no speed or acceleration is defined. For best performance try used a fixed time interval on the position update.

Register Name	Profile Mode = 4	Profile Mode = 5			Description
	CML Registers	Modbus Registers(1)	TCP/UPD Bytes(2)	Ethernet/IP O2T Bytes(3)	
TargetPosition	P1	41029	0-3	0-3	The target position the motor will move to. • Used as an absolute position
TargetTorque	N1	41033	8-9	8-9	The maximum torque the motor will use.
Filter Gain	A1	41035	10-11	10-11	The A1 register is used to store the filter gain.
Controlword	R1	41039	14-15	14-15	Bits in the Controlword dictate the execution. A more detailed description can be found below.
ModeOfOperation	Set with "_mop"	41041	16	16	Sets the mode of operation see Setting Mode of Operation
DigitalOUT	N/A	41043	17	17	Binary value to set the digital output (Register K34=44 to enable this function)

(1) See Modbus TCP for additional details
(2) See TCP/UDP Port 10002 for additional details
(3) See EtherNet/IP for additional details. EtherNet/IP includes AOIs that setup all required mappings.

Controlword Usage

Profile mode uses some specific bits in the Controlword

BIT	Name	Value	Description

B0	Start/New Set Point	0	• Do not execute any changes in target position register • Do not start a new mode • Any active move will be completed

		0 1	• Start a profile move defined by the profile registers
			• 0 1 transition is required to start a profile after
			  ◦ an error has been cleared
			  ◦ a home has been completed
			  ◦ a stop command has been given from CML or an input.

		1	Execute any changes in any of the profile registers immediately.
B1	Halt	0	Do nothing
		1	• Halt the tracking position executed • Prohibit execution of a profile if no profile is being run
B3-B2	N/A		
B7-B4	See Controlword		

Dynamic Position Mode Example

The following example uses the standard registers to continuously update the motor position

Dynamic Position Mode Example

```
K87=10
V1="mop"        //assign V1 to set the mode of operation

a1=10               //set the filter gain to 10 (1-1024)
p1=0               //target position
p2=10               //amount to increment each iteration of the logic scan
N1=1100               //run motor at full torque

|2                       //reset position to 0


L1.1               //Logic bank 1 to init all parameters
R1=0               //clear control word
V1=4               //set mode of operation to 4
|2                       //set motor position to 0
P1=0               //set target position to 0
R1=1               //set control word to start running dynamic position
JL2.1               //jump to L2
END.1

L2.1               //L2 will continuously loop every #ms set in K87.
p1=p1+p2;        //increment target position by P2
end.1
```

Torque Mode

Torque mode would be used where there is no target position but the motor is required to continuously hold or move at a required maximum torque. In this mode the toque set is a maximum torque and the torque will only be reached if required.

Register Name	**Profile Mode = 10**	**Profile Mode = 11**			**Description**
	CML Registers	**Modbus Registers[1]**	**TCP/UPD Bytes[2]**	**Ethernet/IP O2T Bytes[3]**	
TargetSpeed	S1	41031	4-7	4-7	The maximum speed of the profile

TargetTorque	N1	41033	8-9	8-9	The maximum torque the motor will use while moving. • +ve N1 will rotate the motor in a +ve direction (typically CW) • -ve N1 will rotate the motor in a -ve direction (typically CCW)
Controlword	R1	41039	14-15	14-15	Bits in the Controlword dictate torque execution. A more detailed description can be found below.
ModeOfOperation	Set with "mop"	41041	16	16	Sets the mode of operation see Setting Mode of Operation
DigitalOUT	N/A	41043	17	17	Binary value to set the digital output (Register K34=44 to enable this function)

(1) See Modbus TCP for additional details
(2) See TCP/UDP Port 10002 for additional details
(3) See EtherNet/IP for additional details. EtherNet/IP includes AOIs that setup all required mappings.

Controlword Usage

Torque mode uses some specific bits in the Controlword

BIT	Name	Value	Description
B0	Start/New Set Point	0	• Do not follow the N1 target torque • Target Speed S1 is still executed • Do not start a new torque move

		0 1	• Start a torque move defined by the torque registers
			• 0 1 transition is required to start a torque move after
			  ◦ an error has been cleared
			  ◦ a home has been completed
			  ◦ a stop command has been given from CML or an input.

		1	Following N1 target torque
B1	Halt	0	Do nothing
		1	• Halt all motion and inhibit further motion • Halt all torque related parameters (torque & speed)
B3-B2	N/A		
B7-B4	See Controlword		

The interaction of the halt bit and the new set point (nsp) bit is quite specific. See the table below for additional important interaction.

B0 (nsp)	B1 (Halt)	Description
1	0	Motion profile follows torque and speed values
1	0 1	All motion stops
1	1	Motion is inhibited • Changes in speed are NOT followed • Changes in torque are NOT followed
0	1	Operation is the same as above.
0	0	Motion may resume based on values • Torque will NOT be updated • Motion will continue if speed > 0

It is important to understand that the halt bit temporarily halts operation. It does not interrupt motion. This means that a rising edge (0 1) is not required on B0 to continue operation. To keep prohibiting motion the following could be use

- Keep the halt bit at 1
- Set target velocity = 0
- Move out of torque mode

Torque Mode Examples

The following example uses the standard registers to run a torque move. This could be run from a Control Room CML script.

Torque Mode Example

```
R1=0                //clear the controlword before setting the mode
_mop=10               //set the mode of operation to 10. Torque mode
S1=100               //set target speed
N1=300               //set the target torque to 30.0% of rated
R1=1                //start the profile move

//While R1=1 any change in the torque register will execute an immediate change
```