

Myostat Motion Control Inc .
Cool Muscle 1
RT3 Application Note

Logic Bank Notes for Cool Muscle Language



1. Logic Banks

1. Basic Logic Bank

This example shows how we can write a very basic logic bank and execute it. This logic bank does a simple mathematic calculation. If for example the motor is connected to a ballscrew actuator that had a 2mm pitch (2mm/rotation), the motor resolution is 1000 pulses per revolution and you want to have a variable that always changes a position from millimeters to pulses.

CML Code Used:

P1 → calculated position in pulses

P2 → entered position in mm

V1 → screw pitch

V2 → motor resolution

L1 → beginning of logic bank 1

END → end of logic bank

[L1 → execute/run logic bank 1

]L → stop the logic bank

K87 → logic bank execution time in milliseconds.

Example CML Code:

```
/set the scan time
```

```
K87=200
```

```
/set default position to 0
```

```
P1=0
```

```
/position in mm
```

```
/can be changed at will
```

```
P2=50
```

```
/screw pitch
```

```
V1=6
```

```
/motor resolution
```

```
V2=1000
```

```
/calculate the pulses
```

```
L1
```

```
P1=P2*V2/V6
```

```
END
```

```
/start the logic bank  
[L1
```

Notes:

- 1) Once started the equation will return P1 on the serial port. To stop this and only see the value if queried we can put a semi-colon ';' at the end of the line. This stops the logic bank from displaying the result. I.e. $P1=P2*V2/V6$;
- 2) The logic bank can be stopped by sending]L.

2. Starting and Stopping Logic Banks

A logic bank is started with the "[L#" command, where '#' is the logic bank number, and stopped with the "]L" command.

This example application sets V2 equal to the motor position.

CML Code Used:

V1="Px" → internal motor position state
V2 → value to be set to motor position

L2 → beginning of logic bank 2
END → end of logic bank

[L2 → execute/run logic bank 2
]L → stop the logic bank

K87 → logic bank execution time in milliseconds.

Example CML Code:

```
/set the scan time  
K87.1=200
```

```
/internal motor position  
V1="Px"  
/variable to save the position  
V2=0
```

```
/set V2 to position  
L2  
V2=V1  
END
```

```
/start the logic bank  
[L2.1
```

Notes:

- 1) Once the logic bank is executing it will continue to run in the bank ground. In this application example it will execute every 200ms.
- 2) To stop the logic bank send “[L”.

3. Starting a Logic Bank on Power-up

K85 sets the startup/power-up logic bank. E.g. Set K85=1 to run logic bank 1 on power-up.

E.g.

- 1) Set K85=1
- 2) Run the Basic Logic Bank example.

4. ‘if - then - else’ statement in a logic bank

Logical statements can be used with variables, inputs and most other registers inside logic and program banks. This example looks at the status of input 3 and will start or stop the motor depending on if it is on or off.

Logical statement overview:

LOGIC, TRUE, FALSE

E.g. (with V1=0)

CML Code	C Equivalent
I3==V1,].1,^.1	<pre> if (IN3==0) {].1 //stop the motor } else { ^.1 //start the motor } </pre>

CML Code Used:

V1.1=0 → variable set to 0 to compare to Input 3

I3.1 → Input 3 bank status

].1 → CML command to stop a motor

^.1 → CML command to start a direct mode move

P0.1=1000000000 → set direct move position to 1 billion for speed control

S0.1=50 → set direct move speed to 50

A0.1=10 → set direct move accel to 10

L1.1 → beginning of logic bank 2

END.1 → end of logic bank

[L1.1 → execute/run logic bank 2

K87.1=1 → logic bank execution time to 1 millisecond.

Example CML Code:

```
/set the scan time
K87.1=1

/internal motor position
V1.1=0

/set direct mode variables
P0.1=1000000000
S0.1=50
A0.1=10

/set V2 to position
L1.1
I3.1==V1.1,].1,^.1
END.1

/start the logic bank
[L1.1
```

Notes:

- 1) Like most logical statements the compared value can be implied. I.e. you can say I3.1,].1,^.1
- 2) Logical statements can be used in program and logic banks.