

Magellan® MC58113

Series Motion Control ICs



Magellan® MC58113 Series Motion Control ICs provide high performance motion control with an integrated digital current loop. Available as a single, one-axis IC, the MC58113 IC is a programmable device that provides positioning, velocity, and torque control for Brushless DC, DC Brush, and step motors.

A Powerful Motion Controller

Magellan Motion ICs are complete motion controllers requiring only an external bridge circuit or amplifier to be functional. They are driven by a host using either a parallel bus, SPI (Serial Peripheral Interface), CANbus 2.0B, or RS232/485 serial. User selectable profiling modes include S-curve, trapezoidal, velocity contouring and electronic gearing. PID servo loop compensation utilizes a 32-bit position error and includes velocity and acceleration feedforward. High performance FOC (field oriented control) provides high accuracy, ultra-low noise motor operation.

Easy to Use and Program

All Magellan Motion Control ICs provide a flexible and powerful instruction set to initialize and control motion axes, monitor performance, and synchronize overall machine behavior. Working with Magellan ICs, and Pro-Motion® development software makes it fast and easy to graph and analyze system performance; while C-Motion® language allows you to develop your own application using C/C++.

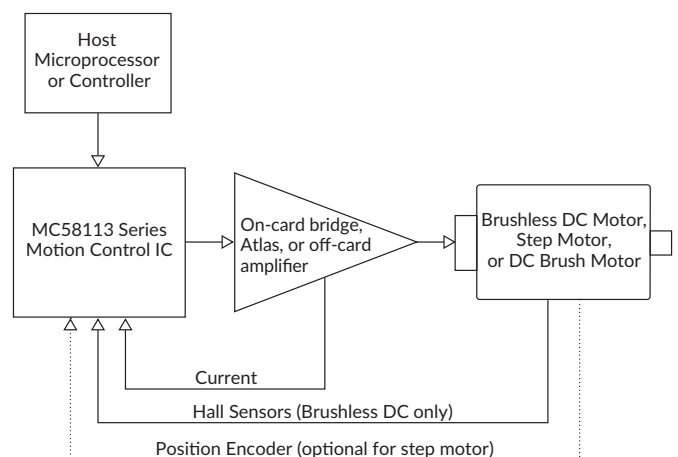
Flexible Offering

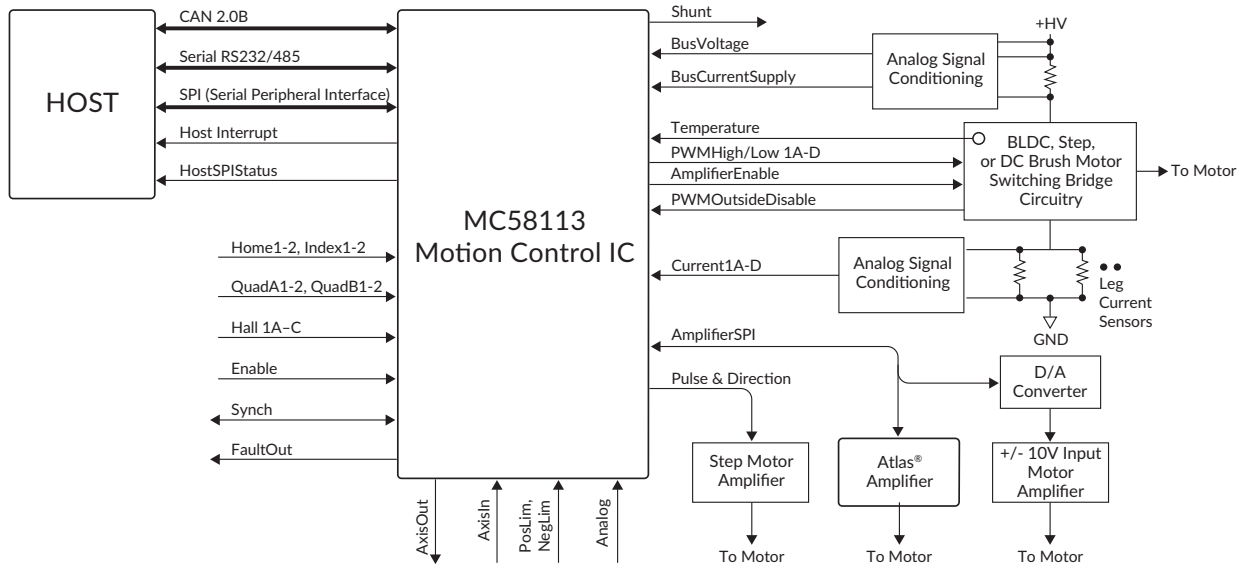
There are four MC58113 Series ICs, each packaged in a single 100-pin TQFP operating at 3.3 V. The MC51113 provides dedicated control of DC Brush motors, the MC53113 provides control of three-phase Brushless DC motors, the MC54113 provides control of two-phase step motors, and the MC58113 can be software selected to provide control of any of these motor types.

FEATURES

- Single axis, single IC
- Position, velocity, and torque control
- Brushless DC, DC Brush, and step motor control
- S-curve, trapezoidal, velocity contouring, and electronic gearing profiles
- SPI (Serial Peripheral Interface), serial RS232/485, and CANbus communications
- 1.5 axes (primary and auxiliary encoder) control
- Advanced PID filter with velocity and acceleration feedforward
- High performance current control of each motor phase
- High/Low switching amplifier control with programmable deadtime and charge pump refresh
- Velocity, position and acceleration changes on-the-fly
- Programmable position loop time from 50 μ sec to 1.1 sec
- FOC (field oriented control)
- Incremental encoder quadrature input (up to 25 Mcounts/sec)
- Synch pin feature allows multiple axes to be synchronized to $<1 \mu$ sec
- Internal motion trace NVRAM for performance optimization
- Overcurrent, over/undervoltage and overtemperature detect
- Directional limit switch, index, and home inputs
- Axis settled indicator, tracking window and automatic motion error detection
- General-purpose analog input
- Programmable dual biquad filters
- Programmable acceleration and deceleration values
- Compact 100-pin TQFP package

CONFIGURATION





MC58113 SERIES SPECIFICATIONS


Parameters	Value
Configurations	MC51113 - DC Brush motor MC53113 - Brushless DC motor MC54113 - Step motor MC58113 - Motor type software selectable
Host communication options	Serial RS232/485 CANbus 2.0B SPI (Serial Peripheral Interface)
Position range	-2,147,483,648 to +2,147,483,647 counts
Velocity range	-32,767 to 32,767 counts/sample
Acceleration and deceleration range	0 to 32,767 counts/sample ²
Jerk range	0 to 1/2 counts/sample ³
Servo loop range	50 usec to 1.1 sec
Position error resolution	32 bits
Commutation rate	20 kHz
Microsteps per full step	Programmable, up to 256
Signal inputs axis 1	QuadA/B, Index, Home, Hall A/B/C AxisIn, Pos/NegLimit
Signal inputs axis 2	QuadA/B, Index, Home
Maximum encoder rate	25 Mcounts/sec
Internal trace buffer	32 KB
Internal NVRAM buffer	2 KB
Operating temperature (Ta)	-40° C to 85° C
Supply voltage operating range (Vcc)	3.0 V to 3.6 V
Package and dimensions	100-pin TQFP, 14 x 14mm

AMPLIFIER CONNECTION OPTIONS

On-board PWM amplifier circuitry	
PWM output rate	20, 40, or 80 kHz
Current control modes	FOC (field oriented control), A/B, third leg floating, voltage mode floating, voltage mode
Current loop rate	20 kHz
PWM output modes	High/Low, Sign/Magnitude, 50/50

External +/- 10V input amplifier	
Amplifier SPI bus serial DAC	16 bits

Pulse & direction input amplifier	
Pulse and direction output rate	up to 1.0 Mpulses/sec

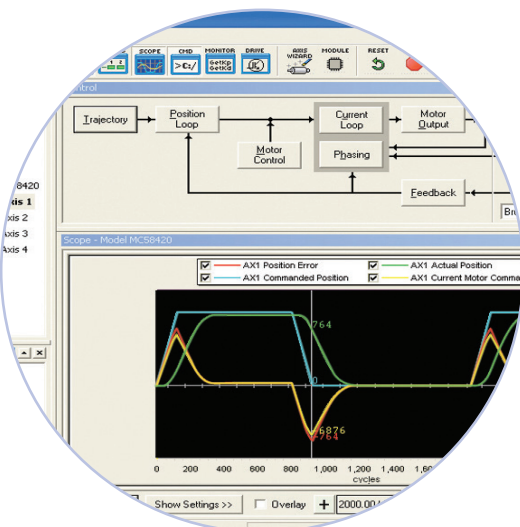
ATLAS® Digital Amplifiers		
Voltage Input	12-56 VDC	
Microstepping resolution	256	
PWM frequency	20, 40, 80 kHz	
Current Loop rate	20 kHz	
Power rating options	75W, 250W, 500W	
Mechanical Dimensions	Ultra Compact size: 1.05" x 1.05" x .53" (27mm x 27mm x 13mm) Compact size: 1.52" x 1.52" x .60" (39mm x 39mm x 15mm)	

Development Tools

1 EASY START-UP Developers Kit

INCLUDES

- MC58113 Developer Kit board
- Pro-Motion software
- Software Development Kit (SDK) with C-Motion
- Complete manual set
- Complete cable and prototyping connector set



2 TUNE & OPTIMIZE Pro-Motion® GUI

Pro-Motion is a sophisticated, easy-to-use Windows-based exerciser program for use with PMD motion control ICs, modules, and cards.

FEATURES

- Motion oscilloscope graphically displays processor parameters in real-time
- Autotuning
- Ability to save and load settings
- Axis wizard
- Distance and time units conversion
- Motor-specific parameter setup
- Axis shuttle performs programmable motion between two positions
- Communications monitor echoes all commands sent by Pro-Motion to the board
- Advanced Bode analysis for frequency machine response

3 BUILD THE APP C-Motion®

C-Motion is a complete, easy-to-use, motion programming language that includes a source library containing all the code required for communicating with PMD motion ICs, boards, and modules.

C-MOTION FEATURES INCLUDE:

- Extensive library of commands for virtually all motion design needs
- Develop embeddable C/C++ applications
- Complete, functional examples
- Supports PC/104, serial, CAN, Ethernet, and SPI communications

```
...code for executing a profile and tracing...
...captured in this example could be used for tuning the Pro-Motion...

SetTraceMode(hAxis1, PMDTraceOneTime);

// set the processor variables that we want to capture
SetTraceVariable(hAxis1, PMDTraceVariable1, PMDAxis1);
SetTraceVariable(hAxis1, PMDTraceVariable2, PMDAxis1);
SetTraceVariable(hAxis1, PMDTraceVariable3, PMDAxis1);

// set the trace to begin when we issue the next update command
SetTraceStart(hAxis1, PMDTraceConditionNextUpdate);

// set the trace to stop when the MotionComplete event occurs
SetTraceStop(hAxis1, PMDTraceConditionEventStatus,
PMDEventMotionCompleteBit, PMDTraceStateHigh);
SetProfileMode(hAxis1, PMDTrapezoidalProfile);

// set the profile parameters
SetPosition(hAxis1, 20000);
SetVelocity(hAxis1, 0x200000);
SetAcceleration(hAxis1, 0x1000);
SetDeceleration(hAxis1, 0x1000);

...tion...;
```