999

softMC
Multi-Axis Controller
Features and Functions

Revision 1.3

softMC 3

softMC 7

SERVOTRONIX
always in motion™

Revision History

Doc. Rev.	Date	Remarks
1.3	Aug.2017	Add softMC 3, update EtherCAT rate
1.2	Dec.2014	Introduction update
1.1	Dec.2014	Revisions
1.0	Dec.2014	Initial publication

Copyright Notice

Disclaimer

This product documentation was accurate and reliable at the time of its release. Servotronix Motion Control Ltd. reserves the right to change the specifications of the product described in this manual without notice at any time.

Trademarks

All marks in this manual are the property of their respective owners.

Contact Information

Servotronix Motion Control Ltd.
21C Yagia Kapayim Street
Petach Tikva 49130, Israel

Tel: +972 (3) 927 3800
Fax: +972 (3) 922 8075

Website: www.servotronix.com

Technical Support

If you need assistance with the installation and configuration of the softMC drive, contact Servotronix technical support: tech.support@servotronix.com

Contents

Introduction

The softMC is a multi-axis motion control software and hardware package that provides extensive programming capabilities for a variety of automation and robotic applications.

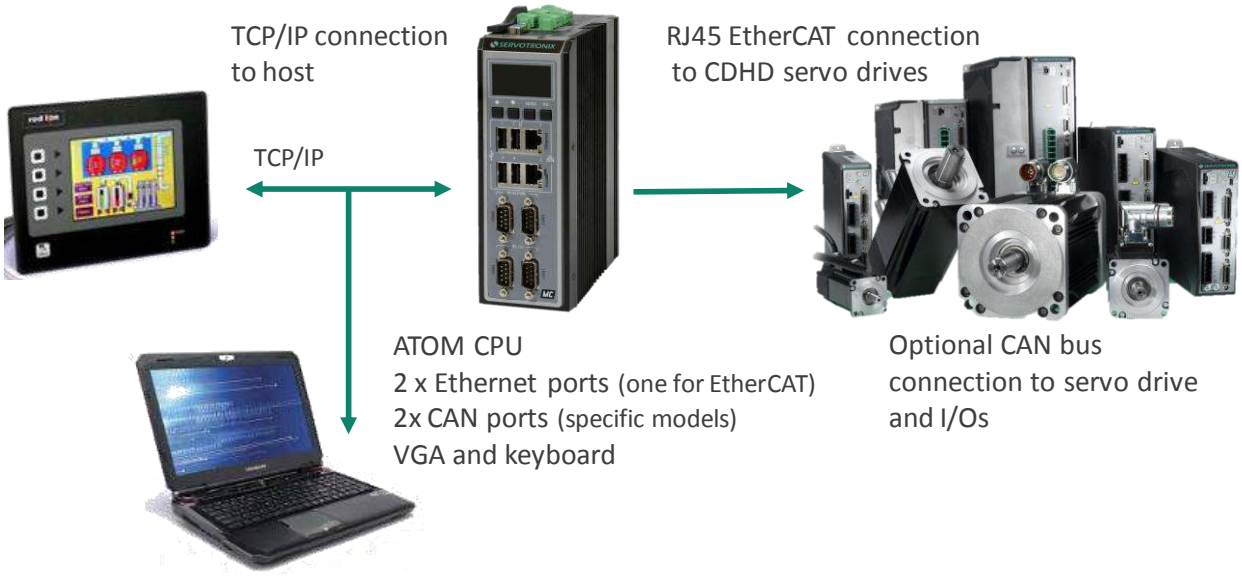
Operated by Linux, with real-time extensions, the softMC runs on a qualified industrial PC, providing an open and modular machine control environment. Its rich programming language allows complete flexibility to create motion programs, with support for pre-emptive multi-tasking and asynchronous event response.

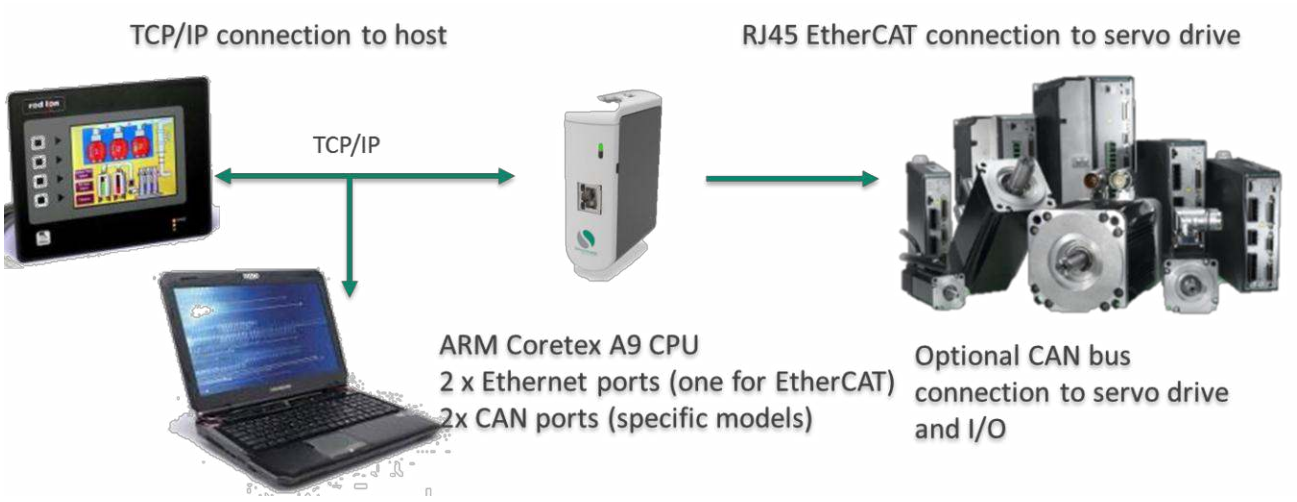
Extensive motion Cartesian and robotics functionalities support standard mechanics and robot types such as XY, XYZ, DELTA, PUMA, SCARA, palletizing cooperate as well as other non-standard robotic kinematics.

This document describes the features and functions of the softMC motion controller, organized according to the various categories of controller functionality.

Servotronix's Complete Motion System – from Controller to Encoder

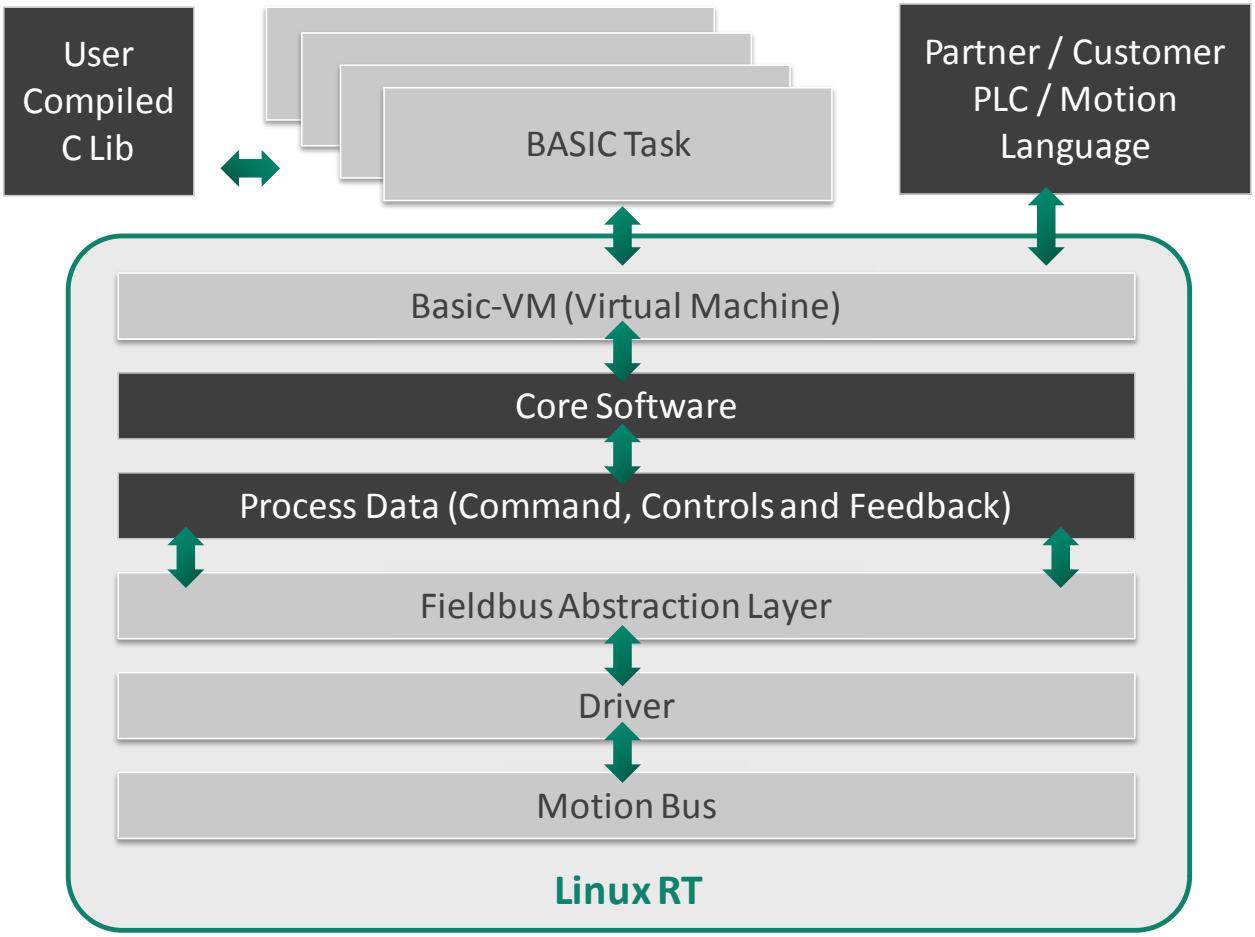
The softMC is designed to integrate with Servotronix's servo and stepper drive-motor systems to provide a complete motion solution in a cost effective package:





Note: we use same firmware for softMC 7 and softMC 3

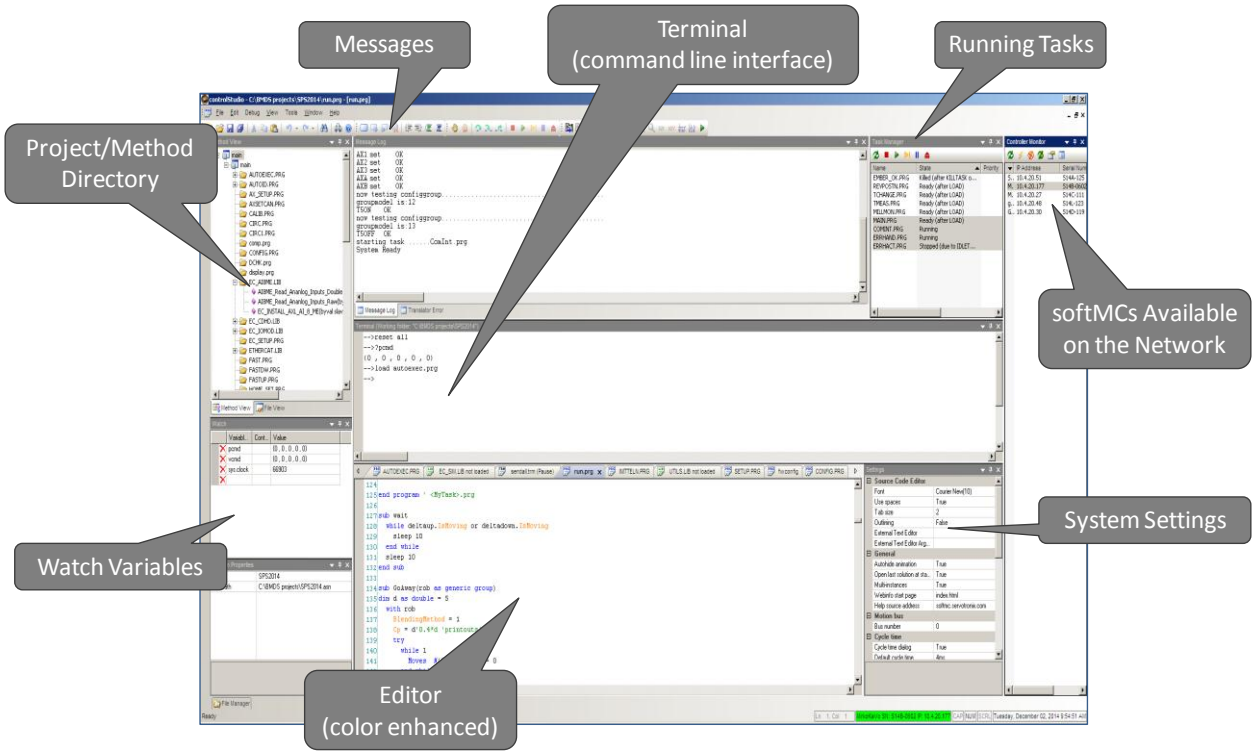
SoftMC Software Architecture



SoftMC Features and Functions

Program Development Environment

Integrated Development Environment (IDE)	**ControlStudio** ■ Windows-based text editing program ■ All programs are plain ASCII files ■ Free of charge



Language

MC-Basic	■ Based on standard Microsoft Basic, and extended for sophisticated motion control
Flow Control	■ Conditional statements Standard If-Then-Else and Select-Case structures ■ Loops For, While and Until constructs ■ Mathematical Full support of all rational and trigonometric functions ■ Array handling Multi-dimensional arrays (matrices) Built-in protection of index over-under run

Functions and Subroutines	■ Argument passing By value By reference Recursion ■ Global and local functions
Data Types	■ Standard data types: Integers (32-bit) Double Structures Dynamic strings (unlimited size, UTF-8) ■ Specialized data types: Axes Groups Points (robotics) Cam tables PLS Compensation tables
Built-in Functions	■ Trigonometric and rational math functions ■ Logic operations (bitwise and integer) ■ Frame operations (robotics) ■ Direct and inverse kinematics as math functions
Libraries	■ Local libraries Functions and subroutines used only in selected tasks ■ Global libraries Functions and subroutines used throughout the system as an extension of the language
Error Handling	■ Inline error handling ■ Catch and throw ■ User defined errors and notes ■ Permanent error history, with time and date of each error ■ System error handler ■ Local error handler
Event Handling	■ Real-time system events as user-defined program interrupts ■ Selectable priority level
Multitasking	■ Up to 256 different tasks ■ 16 priority levels in preemptive, round-robin scheduling ■ Multiple instances of same programs as different running tasks ■ Starting, stopping and idling a task from another tasks ■ Semaphores for multitask manipulations
Extensions	■ C/C++ externally compiled modules can be combined within MC-Basic written tasks

Motion Control

Interpolation	■ Multi-axis interpolation System supports up to 64 axes Axes can be configured to work together as a group Groups of 2 or more axes can be interpolated together Some axes can be simulated (to serve as a virtual master) while others are actual ■ Single axis Move – point to point motion Jog – endless constant velocity motion ■ Groups Move – multi-axis synchronized point to point move with cruise velocity Circle – circular or helical motion of 2-axis or 3-axis groups ■ Motion start Interrupting current motion (on-the-fly changes) Waiting for motion to be settled Waiting for motion to end ■ Motion Profiles Trapezoidal velocity Sine acceleration Trapezoidal acceleration ■ On-the-fly changes of velocity Velocity of an axis, group or system can be reduced or increased online ■ Stop/Proceed functionality Axis/group can be stopped and the stopped motion can be reconstructed using Proceed command capabilities ■ Attachment mechanism provides exclusive control of one task over an axis or group
Camming and Gearing	■ Up to 65536 cam tables ■ Interpolation Linear interpolation 5-degree polynomial ■ Linkage Cam tables can be linked together as a double-linked tree ■ Cyclic cams Defined number of cycles Unlimited number of cycle ■ Absolute cams ■ Cam master offset ■ Camming/gearing sources Another axis position command Any axis position feedback Any axis external position sources (if supported by the drives)

Flying Saw	■ Flying saw synchronizes a master axis with a slave axis at a certain sync position. ■ MC-Basic programming enables engaging/disengaging of camming, and setting of parameter values. ■ Master axis can be real or simulated, and use external position encoder of the drive. ■ Cam table defined by an array of points using linear interpolation of 5-degree polynomial. ■ Cyclic or one-pass tables. ■ Translation of the camming curve using master and slave offset parameters. ■ Cam tables can be switched and exchanged online.
User Units	■ User definable units ■ Position, velocity, acceleration and jerk units are accessible by the user and can be defined in different ways (e.g., position, degree, velocity, rpm)
Coupling	■ Simple kinematics A group axis can be used via coupling matrix that defines joints. ■ Joints are virtual axes that represent a linear combination of two or more real axes. ■ Useful in mechanical coupling or orthogonal correction of XY tables.
Compensation	■ Supports extreme accuracy requirements Up to 65536 different compensation tables 1, 2 or 3 dimensional Linear, bilinear interpolation between the points Loadable from binary files Compensation tables can be added to each other (superposition) ■ Compensation setups One-to-one Cross axis compensation 3D-3D compensation ■ Variable backlash Direction dependent compensation tables ■ Backlash compensation Adding constant backlash compensation value depending of the axis movement direction

Programmable Limit Switches	■ Defined as data types Enable language manipulation as any other system variables ■ Different sources Can be triggered on position command or position feedback ■ Extensive features Repetition Hysteresis Polarity Delayed action ■ Connectivity Works on any digital output defined in the system ■ Fast PLS Microsecond accuracy using interpolated (2^{nd} degree) trigger position ■ Path PLSs 2 or 3 dimensional triggering
Robotic Models	■ Robotic models Wide range of standard and user defined robot models ■ Interpolation Move – joint interpolated motion Moves – Cartesian straight line interpolation Circle – Cartesian circular or helical interpolation ■ Open chain models SCARA PUMA SpeedPicker ■ Parallel robots DELTA Scissor kinematics ■ Inverse dynamic model Dynamic model for torque computation Automatic dynamic parameter identification (e.g., mass, inertia)of selected robot models ■ User access to all geometric parameters of every robot models Link lengths Gear rations Coupling

Robotics	■ Motion frames: Tool Base Machine base Workpiece ■ Configuration flags Lefty/Righty Above/Below Flip/NoFlip Singularity handling ■ Conveyor tracking Picking and placing object from a moving conveyor ■ Cartesian gearing External axis geared proportionally to the traveled tool-point path ■ Point data type Location variables tailored for each robot model (X, Y, Z, Yaw, Pitch, Roll), and manipulated by standard arithmetic operations (+ - * /) and compound operator (matrix multiplication) ■ MC-Basic commands for off-line manipulations: ToJoint() - Inverse Kinematics ToCaty() - Direct Kinematics
Blending	■ Continuous Path (CP) Continuous velocity during blending phase with assured limits of acceleration and velocity; blending starts at a specified target point distance. ■ Super Position (SP) Blending condition at a given percentage of distance from the target point. ■ Advanced Interpolation (AI) Taking a large number of points into account in advance. Useful for implementing CNC look-ahead buffers. Buffer can be streamed online or given completely in advance. ■ Various Advanced Interpolation (AI) methods Via Through Smoothing

Synchronization	■ Synchronizing different axes/groups Motion starts and finishes at same time ■ Moving frame data object For synchronization of different robots (master-slave) ■ Various kinematic models of moving frames: Linear Rotary Different robot ■ Conveyor tracking Picking and placing objects from a moving linear or rotary platform Working window defined by upstream and downstream master points Triggered by the captured position entering working window
Dynamic Models	■ Model identification Automatic identification of model parameters, including mass, inertia, and coefficients of viscous and Columbus friction ■ Torque injection Improving control behavior and reducing settling time ■ Collision detection Monitoring torque error allows reaction to collision much earlier than when using position error

Diagnostics

Simulation	■ Individual axis simulation Each axis can be individually switched from real to simulated mode without losing any performance as compared to a real axis. ■ Complete system simulation The entire softMC system can be run in a simulated environment (Virtual Box) on your laptop.
Recording	■ Realtime recording Any numerical expression, even complex motion properties and system variables, can be recorded in realtime; e.g., at every motion bus sample. ■ Record Viewer Record files can be retrieved from the softMC and analyzed (zoom, derivation, scale, 2D plots) as part of standard ControlStudio functionality ■ Online plotting ■ Changes in variables can be observed online in realtime

Debugging	■ Watch window Querying of local (task) variables in Terminal window Querying of global variables in Terminal window ■ Task debugging StepIn function StepOut function Breakpoints ■ Task Manger Showing state, priority, and execution line of every task.

Motion Bus

EtherCAT	The EtherCAT interface system is a fast, vendor-independent, Ethernet-based, realtime open network for servo and I/O communications that works with CANopen over EtherCAT (CoE) servo drives and I/O devices. ■ RJ45 Ethernet cables are plugged directly into the Ethernet port on the PC, and servo drives are connected in a simple, single-line daisy-chain; no hardware is required. ■ Up to 64 servo drives (plus additional I/O stations) can be integrated in one network. ■ The EtherCAT interface system is fast, with an interpolation rate as fast as (Cycle time) **softMC 3** (ARM Cortex A9): 2ms, 4ms, 8ms **softMC 702** (PC Atom): 1ms, 2ms, 4ms, 8ms **softMC 705** (PC CORE i5) 0.5ms, 1ms, 2ms, 4ms, 8ms Note: interpolated rate depend on number of axis, profile type PDO setting and can be measured by CPU real time load which should not exceed 50% ■ Simplified configuration and standard cabling, and elimination of adapter cards, switches and hubs. ■ Economical system, in terms of both initial investment and maintenance costs. ■ EtherCAT: CoE/FoE/EoE, distributed clock. This interface system is based on 100 Mbps Ethernet. ■ Major EtherCAT functions: CoE, FoE, DC Sync, Line/Star/Ring Topologies, Hot Connect, Network Management API to monitor and control slaves
CANopen	■ **CiA 301**: CANopen Application Layer and Communication Profile ■ **CiA 305:** CANopen Layer Setting Services (LSS) and Protocols ■ **IEC 61800-7-1**: Interface Definition; (previously CiA 402-1: General Definitions) ■ **IEC 61800-7-201**: Profile Type 1 (CiA 402); (previously CiA 402-2: Operation Modes and Application Data) ■ **IEC 61800-7-301**: Mapping of Profile Type 1; (previously CiA 402-3: PDO Mapping)

Connectivity

Ethernet (TCP/IP)	■ TCP/IP communication ■ OPENSOCKET Creates a TCP/IP socket and puts socket descriptor into the handle specified by the user. ■ ACCEPT (server) Binds socket to specified port and waits for connection. ■ CONNECT (client) softMC requests a connection from remote host according to IP address and port. ■ PING Used to verify that a remote host can be accessed; remote host must support "ICMP echo request". ■ SET IPADDRESS Used to set/get controller IP address. ■ UDP FastData Streaming update of system state
Modbus	■ Controllers using Modbus communicate via a master/slave relationship. ■ Message structure is a 10-bit packet consisting of: Device address Function code 8-bit data bytes Error checking ■ softMC fully supports Modbus as a master or slave — in TCP, RTU or RTU multidrop configurations. ■ Modbus slave protocol commands are built-in functions on the Ethernet, RS-232 and RS-485 ports. Setting the IP address for Ethernet and the node address for the RTU is the only setup required. ■ Master and slave operation Each communications port can be configured independently for either master or slave operation.
RS232	■ 3 user and1 system standard COM ports ■ Access through standard BASIC channels using: OPEN, PRINT , INPUT, CLOSE ■ Can be extended using USB ■ Baud rate - baud rate of the device set to a specified value. ■ Parity – enable/disable parity detection; when enabled, parity is odd or even. ■ Data bits – number of data bits. ■ Stop bit – number of stop bits. ■ Xonoff – sets raw mode or ^S/^Q flow control protocol mode (optional parameter disabled by default).

File System	■ Full set of BASIC file commands ■ Both binary and text files ■ Accessible from all user tasks ■ OPEN command (read/write binary/text) ■ PRINT command for standard write (strings or numeric) ■ INPUT for reading file ■ TELL/SEEK for fast access ■ Access to both Flash disk and RAM disk

softMC

Features and Functions

Servotronix - 21C Yagia Kapayim St.
POB 3919 Petach Tikva 49130, Israel
Tel: 972-3-927-3800
info@servotronix.com
www.servotronix.com