

GCode Interpreter/KMotionCNC

New Trajectory Planner Option to allow Concave Corners with radius compensation on. It isn't possible to follow an internal corner path with a round tool. This option allows concave corners with the tool following the path as close as possible without gouging. To use enable Tool Setup | Trajectory Planner | Allow Concave Corners. Example GCode file AllowInsideCorners Zig Zag.ngc

Fix issue with GCode Files in UTF-8 coding with percent character that loops back to beginning.

KMotionCNC

Fix issue with Tool Setup | Trajectory Planner Screen with mm/inch cells having negative values not updating when changed.

Fix issue with cases of microscopic moves while Jogging potentially rounding to zero velocity causing error messages.

Pick & Place Example Screen '8 Axis CEMEX PAP' added including button graphics added.

Warning added for duplicate Tool Table entries with same Tool ID

Fix issue with no icon in title bar.

KFLOP to PC Commands added to read/write Tool Table Comments. See new example "Tool Comments.c"

```
void SetToolComment(int gather_offset, int ToolIndex, char *s)
void GetToolComment(int gather_offset, int ToolIndex, char *s)
```

KFLOP to PC Commands added to read Current line of GCode. See new example "KFLOPtoKMotionCNCGetGcodeLine.c"

```
void GetGCodeLine(int gather_offset, char* s)
```

KFLOP to PC Commands added to read current Date and Time. In format: Fri, Apr 03, 2026, 10:48:22

See new example "KFLOPtoKMotionCNCGetDateTime.c"

```
void GetDateTime(int gather_offset, char* s)
```

KFLOP to PC Commands added to read/write Trajectory Planner Velocities, Acceleration, and Resolution.

See new example "GetSetTPParameter.c"

These are now defined in KflopToKMotionCNCFunctions.c

```
// Axis indices
#define AXIS_X 0
#define AXIS_Y 1
#define AXIS_Z 2
#define AXIS_A 3
#define AXIS_B 4
#define AXIS_C 5
#define AXIS_U 6
#define AXIS_V 7
#define NUM_AXES 8
```

```
// Parameter type indices
#define PT_VEL          0
#define PT_ACCEL        1
#define PT_COUNTS_PER_INCH  2
#define NUM_PARAM_TYPES  3

// Set a Trajectory Planner Parameter given Type and Axis
int SetTPParameter(int Type, int Axis, double value)
// Get a Trajectory Planner Parameter given Type and Axis
int GetTPParameter(int Type, int Axis, double *value)
```

KFLOP Commands to set Feed, Rapid, Spindle Overrides (FRO, RRO, SSO) were limited to 2 decimal places. Values less than 0.01 would round to zero. 3 decimal digits are now accepted so 0.001 is accepted and used. 0.001 can also be entered on the KMotionCNC Screen. Previously 0.1 was the smallest allowed value.

Motion Buttons (Jog Buttons) now allow individually, forced disable/enable via setting the Screen Script Var parameter to 0=Force Disable, 1=Force Enable, -1=dynamic automatic behavior based on Job Running or not.

Axis DRO number formats may now be specified as the Text Parameter in the Screen Editor. If Text is specified, then it is treated as DRO format of number to be displayed. Include any axis characters if desired. If different formats for Inches and mm are desired specify both separated them with ; // e.g. X:%10.4f;X:%10.3f

New C Program Example to obtain the current Tool Slot and ID GetCurToolSlotAndID.c

KMotionCNC Screen Editor

Now loads Resource.h and KMotionCNC.rc from either the standard install folder or from the same folder as the executable in case the user has moved binaries to non-standard locations.

Screen file backups are now made in the same folder as the current loaded Screen file rather than always in the standard installed location for Screens.

Axis DRO number formats may now be specified as the Text Parameter in the Screen Editor. If Text is specified, then it is treated as DRO format of number to be displayed. Include any axis characters if desired. If different formats for Inches and mm are desired specify both separated them with ; // e.g. X:%10.4f;X:%10.3f

KMotion

Allow KMotion Installer to run in an ARM Architecture for case where a Virtual Box/Emulator is used.

Change icon to be more visible with Windows in Dark Mode

Kogna

Fix issue with intermittent ADCs conversions and not always timely updating.

Fix issue with boot hanging with bad FPGA not responding properly.

Adds a latched register for the 24 differential inputs. An FPGA 32-bit register holds the latched inputs in the lower 24 bits. The register is defined as:

```
#define LatchedDiffIn 0x1c0 // FPGA Address for 24 Latched Diff Inputs, read as 32-bit word,  
write 0's to clear[/code]
```

The latched bits will be set on the rising edge of the input. Any pulse greater than 60ns should be reliably detected. Write 0's to bits to clear. Note it takes several FPGA 16.7MHz clocks to synchronize and set the latches so reading the register immediately (within 120ns) after the rising edge may not yet show the latched status. See new example C Program LatchedDiffInputs.c

Added ResetEthernet() function to reset Ethernet Transceiver and Ethernet Software Stack.

New Virtual COM Terminal commands: 'E' Reset Ethernet 'R!' Reboot

Kinematics/Geocorrection

Kinematics and Geocorrection had a maximum range of 100 inches by default caused by maximum iterations of 1inch with a maximum number of iterations. MaxCorrPerIteration parameter can now be specified in the Kinematics.txt file. For example MaxCorrPerIteration=100 by 100 iterations will allow 10,000 inches of range. The correction should be small enough such that for the nonlinear system the increment will follow the gradient and not jump to a completely different region in the solution space. Most systems are not so nonlinear for this to be an issue.

Which Kinematics to use, if a \KMotion\DataKinematics.txt is specified now requires an exact name match. Previously first Kinematics with a partial match would be selected. Eg. "Scara" would match "KinematicsScara" as well as KinematicsScara3Axis".

SimpleFormsCS – example C# program using .NET Library

Changed to show creating a KM_Controller object for a 2nd Kogna by IP Address